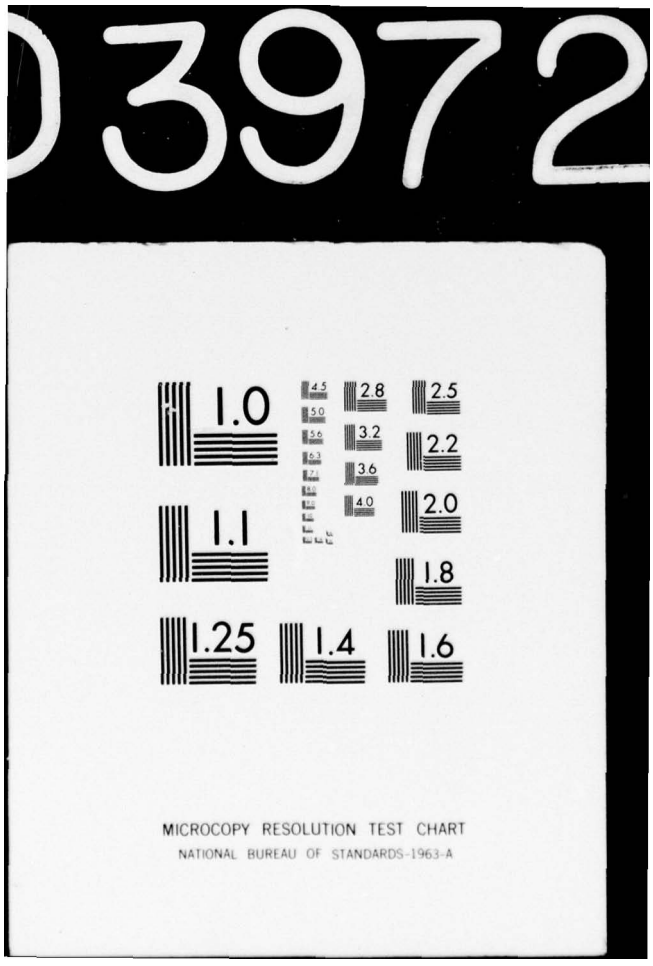
AD-A039 720 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 17/2
MODIFICATION OF THE AN/TGC-26 MULTIPLE ADDRESS PROCESSING UNIT --ETC(U)
MAR 77 R G SHIVELY

UNCLASSIFIED AFIT/GE/EE/77-3 NL

1 OF 2

AD A039720


03972
1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
7.1
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD A039720

B.S.

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

AD NO.
DDC FILE COPY

DDC
RECEIVED
MAY 23 197
B

UNITED STATES AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

Approved for public release; distribution unlimited.

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

GE/EE/77-3

(1)

MODIFICATION OF THE AN/TGC-26 MULTIPLE ADDRESS PROCESSING UNIT TO ACCEPT JANAP 128 FORMAT

THESIS

Robert G. Shively
Captain, Signal Corps
GE/EE/77-3

ACCESSION for
NTIS White Section
DDC Buff Section
UNANNOUNCED
JUSTIFICATION
BY
DISTRIBUTION/AVAILABILITY CODES
Dist. AVAIL. and/or SPECIAL
A 23

DDC
RECEIVED
MAY 23 1977
B

Approved for public release; distribution unlimited.

Note to DDC:

Some of the pages at end of this thesis are fairly light. However, the thesis is completely understandable without them. So please print even if they are hard to read. This is our best copy.

Thanks,

AFIT/GE/EE/77-3

(14)

(6) MODIFICATION OF THE AN/TGC-26 MULTIPLE ADDRESS PROCESSING UNIT TO ACCEPT JANAP 128 FORMAT.

(9) Master's THESIS,

Presented to (1) the Faculty of the School of Engineering (3)
of the Air Force Institute of Technology,
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

(10) Robert G. Shively
Captain, Signal Corps

Graduate Electrical Engineering

(11) March 1977

(12) 132 p.

Approved for public release; distribution unlimited.

012 225

PREFACE

As an Army officer looking for a thesis topic at the Air Force Institute of Technology, I was hard pressed to find a problem relevant to my career in the Signal Corps. The opportunity to work on the modification of a piece of Air Force communications equipment was the perfect answer to my dilemma. The chance to get my hands on a real piece of hardware while meeting and working with the communications personnel of a sister service was as rewarding an experience as the successful results that I achieved.

I would like to thank Major Duane Reynolds of the Air Force Communications Service who, as a former AFIT graduate, was a very cooperative sponsor. Captain Pete Miller took on the role of faculty advisor during a very busy time in his career, and I owe him a debt of gratitude. The men of the 253rd CMBTCG at Wellesley, Massachusetts provided all the assistance and expertise that I required in tackling the MAPU problem.

Most of all, I would like to thank my wife Linda for persevering through 18 months of AFIT, the joys and sorrows of my thesis, and the bulk of the typing of this document.

Robert G. Shively

CONTENTS

CONTENTS

LIST OF FIGURES

LIST OF FIGURES

LIST OF SYMBOLS

BRCBN	DLS signal name for BROADCAST button.
CR	Baudot Code carriage return character.
CRLF	Two carriage returns and one line feed character as a fixed combination.
DETCO	DLS signal name for DET CODE signal.
DREAD	DLS signal name for DELAYED READ signal.
FG	Baudot Code figures character.
JKMS	J-K master slave flip-flop.
LF	Baudot Code line feed character.
LT	Baudot Code letters character.
RESCL	DLS signal name for REST CLOCK signal.
SDET	DLS signal name for STOP DET signal.
SDIST	DLS signal name for STOP DISTR signal.
SELC1-6	DLS signal name for ITEM SELECT buttons one through six.
SP	Baudot Code space character.
SR	Set-Reset flip-flop.

ABSTRACT

The Multiple Address Processing Unit (MAPU) of the AN/TGC-26 tape relay center was originally designed to process ACP 127 message format. With the advent of JANAP 128 as a standardized message format a pressing need was felt for a MAPU modification to allow processing of this format. Under the sponsorship of AFCS/DOOT, a partial MAPU was assembled at the Air Force Institute of Technology, and the code recognition logic circuitry was redesigned. Software simulations of the proposed redesign were tested using the Digital Logic Simulator program on the CDC CYBER computer system. A final prototype version of the modified MAPU was successfully tested at Wellesley Air National Guard Station, Massachusetts, using personnel and equipment of the 253rd CMBTCG. A final schematic diagram and production recommendations are presented herein.

I. INTRODUCTION

Background

The AN/TGC-26 is a transportable tape relay center which was procured by the Air Force in 1973 to support the Tactical Air Control System (TACS) mission through the Air Force Communications Service's active duty and Air National Guard Combat Communications Groups (CMBTCG) (Ref 4:1). Six of these relay centers were procured; two of them are presently in active units and four in the Air National Guard. The AN/TGC-26 tape relay, as shown in Fig. 1, was developed to fulfill the requirements for a slow speed teletype relay function used for transmission and reception of teletype traffic within the TACS and between the TACS and external agencies. The AN/TGC-26 is a shelterized, transportable switching complex designed to receive encrypted messages, convert them to unencrypted paper tape, segregate addressees as necessary, re-encrypt them, and transmit them to the proper addressees.

Although much of the electronics within the complex is integrated circuitry, it is a manual facility in that all message logging and routing is done by Air Force communications personnel. The complex contains the equipment required to produce paper tape copies of incoming messages, to duplicate copies for individual routing, and to transmit them on outgoing circuits. Receive Reperforators (tape punches) are

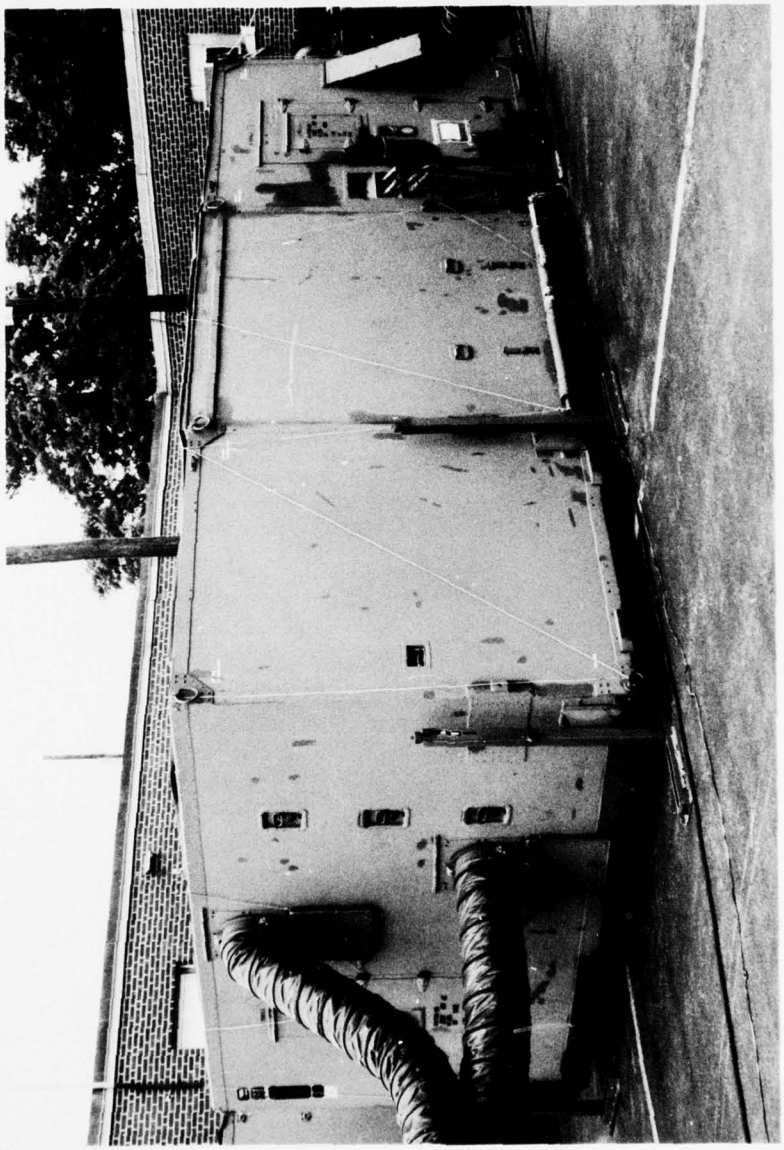


Fig. 1. AN/TGC-26

used to convert an incoming signal to paper tape at 100 words per minute. The Multiple Address Processing Unit (MAPU) duplicates a multiple addressed message by producing singly addressed copies. Transmitter/Distributors (TDs) convert the tapes back into electrical signals for transmission.

Multiple Address Processing Unit. The MAPU, shown in Fig. 2, is a vital component of a tape relay center. The AN/TGC-26 MAPU fits standard 19" relay racks and stands four feet high. It contains one transmitter/distributor and six reperforators. A tape fed into the transmitter/distributor reproduces a preselected number of tapes, each tape containing one or more routing indicators (addresses). Hardware, mostly Small Scale Integration (SSI) and Medium Scale Integration (MSI) circuitry, is used to perform the necessary logic functions. A front panel view of the MAPU is shown in Fig. 3.

Problem Statement

The MAPU logic circuitry, as originally designed, handles messages in the ACP 127 military message format. A message format is simply a standardized way of organizing the basic components of a message: the heading, the routing indicators, and the text. ACP 127 is no longer used by the United States armed services; consequently, the AN/TGC-26 MAPU has extremely limited usefulness. The MAPU needs to be redesigned to handle the current message format (JANAP 128) used by all United States forces. Since some of our NATO allies still use ACP 127, the capability of handling the older format must be retained. Such a change requires redesign of the MAPU input logic circuitry.

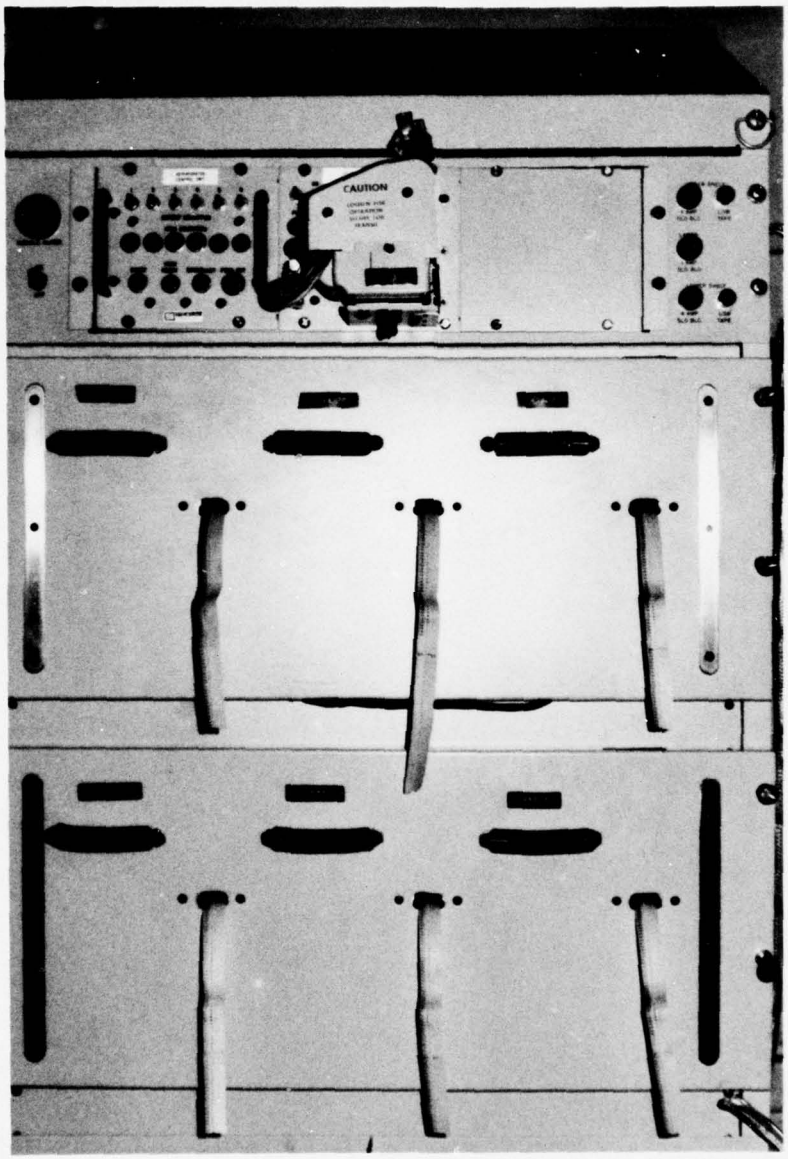


Fig. 2. Multiple Address Processor (Full View)

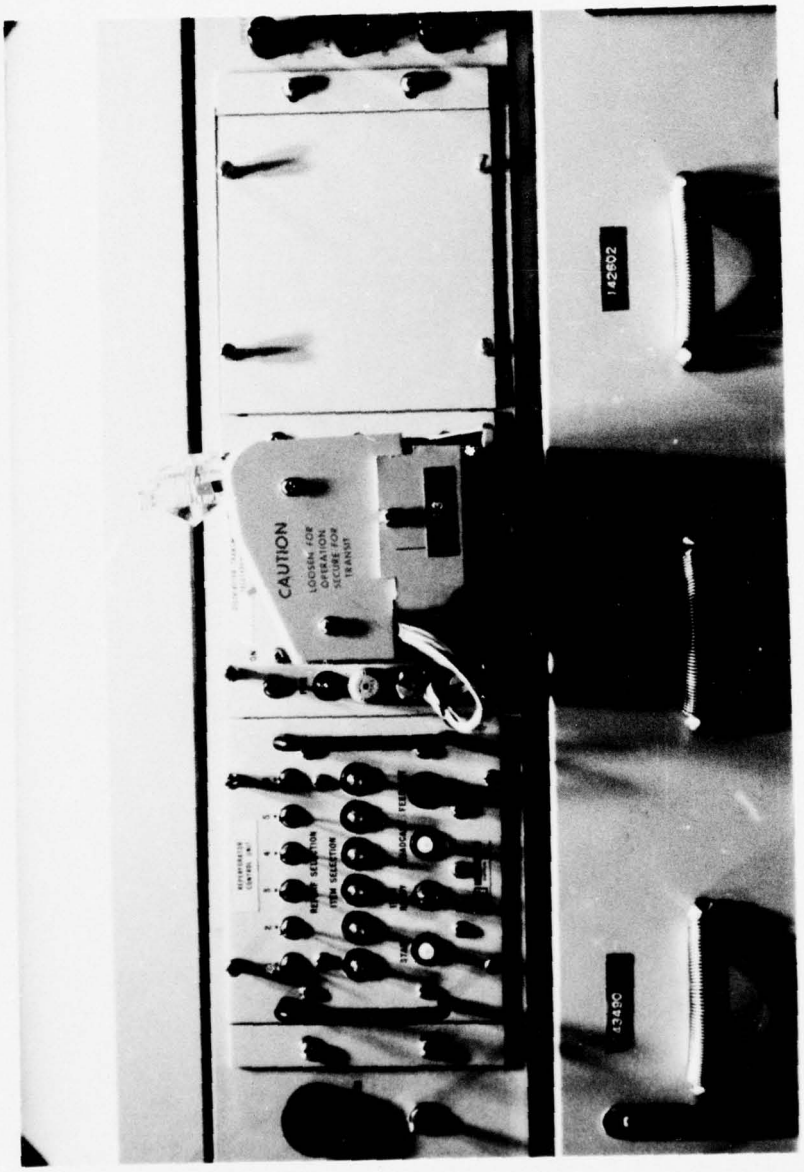


Fig. 3. Multiple Address Processor (Control Panel)

ACP 127 Operation. As already mentioned, the MAPU was designed at a time when the message format used was ACP 127. Since all transmitted messages were in a standard format, it was possible to design the MAPU to recognize certain character combinations as flags to indicate halts and mode changes. Since paper tape transmission uses a five bit Baudot code for all characters, such recognition capability can be built using logic circuitry. Fig. 4 is an example of a multiple addressed message formatted in ACP 127. The positions where the MAPU halts or changes operating mode are shown in circled lower case letters.

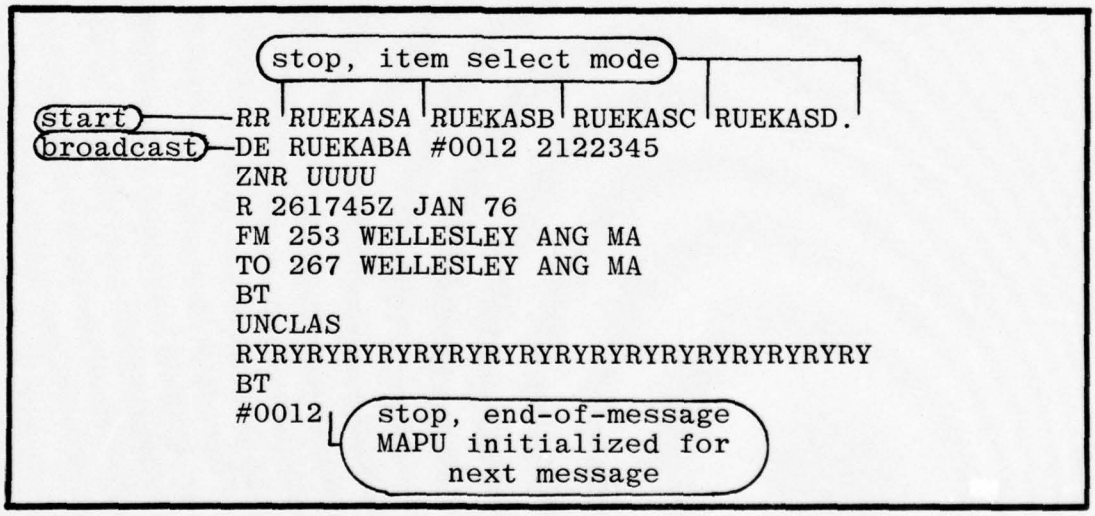


Fig. 4. ACP 127 Format

To segregate routing indicators with the MAPU, the operator places the master tape of Fig. 4 above in the transmitter/distributor of the MAPU. There are four routing indicators on this tape; therefore the operator activates reperforators one

through four by means of the toggle switches on the MAPU front panel (see Fig. 3). When the start button is depressed, the message heading up to the first indicated stop is duplicated on all four slave tapes. When the first halt code is detected, the MAPU stops, switches to item select mode, and illuminates the ITEM READY lamp. The operator then depresses the ITEM pushbutton associated with the reperforator on which he wants that routing indicator to appear. The MAPU transmits the seven letters of the routing indicator and stops at the next flag halt. This sequence is repeated until the operator has transmitted each of the routing indicators to the desired reperforator. The MAPU stops at the end of the final routing indicator when the halt code is detected, and again illuminates the ITEM READY lamp. The operator now depresses the BROADCAST pushbutton and the following events occur: the ITEM READY lamp is extinguished and the BROADCAST lamp is illuminated; the MAPU logic activates an internal interlock inhibiting the detection of any further flags; the MAPU starts simultaneous transmission of the remainder of the message to all activated reperforators. Transmission of message text continues until the MAPU receives the end-of-message signal from the transmitter/distributor. This signal is initiated when the end of the tape passes through the transmitter/distributor (Ref 1:3-6). The results of the operation just described are as follows: each of the four message tapes produced has the message heading of the master tape, followed by just one of the routing indicators, and then by the full

text of the message from the point of BROADCAST mode to the end-of-message.

JANAP 128. A problem arises when a message format other than that for which the MAPU is designed is used on the master tape. Several years ago all military services, including the Air Force, converted to the use of JANAP 128 message format in order to be compatible with the Autodin high speed message system. An example of the message of Fig. 4 written in JANAP 128 format is shown in Fig. 5. Once again the stops and mode changes are shown in lower case, circled letters.

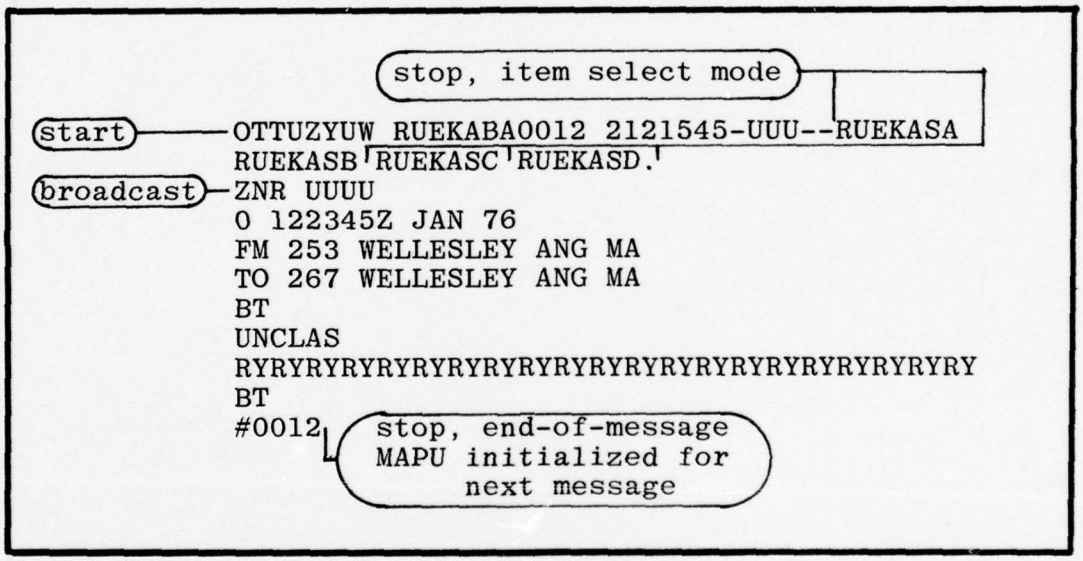


Fig. 5. JANAP 128 Format

The indicated stops and mode changes are the ones which must be designed into the MAPU logic circuitry. In comparing the two message formats it should be noted that while the original design searches for a SP or a CR character in each case, the redesign for the new format is more difficult. In JANAP 128 format there is no common flag for machine halts. The first

routing indicator occurs after two dashes and a LT character. A CR alone cannot be used as a flag for the second routing indicator because the end of line sequence in all teletype messages is two CRs and one LF. The new design needs a memory capability in order to recognize a LF only after two CRs have been detected. The design also needs the capability of looking ahead to recognize this situation and substitute a SP for the original three characters. The design would retain the SP character as a flag since it is common to both formats. The period terminating the routing indicator sequence is critical in Autodin interfacing, and the message would be rejected without it. Therefore the design must be capable of inserting a period after the last routing indicator on each slave tape.

Scope of the Design and Realization

The sponsoring agency for this MAPU redesign is the Air Force Communications Service. As the sponsor AFCS has set forth a number of general design specifications to be met in redesigning the circuitry. These specifications include:

a. The redesigned MAPU must be capable of accepting both ACP 127 and JANAP 128 formatted messages.
b. The modification must mount on or in the present MAPU housing.
c. All components must be available through military supply channels or from commercial sources.
d. All components must meet military specifications.

e. Maximum use should be made of the original MAPU circuitry and components.

f. A per item production cost limit of $1,000 should be kept in mind during the realization phase.

The 253rd CMBTCG in Wellesley, Massachusetts is one of the Air National Guard units which employs the AN/TGC-26. This unit is geographically closest and its personnel are well qualified to comment on MAPU operational problems. After a preliminary meeting with representatives of the 253rd and an opportunity to watch the MAPU in operation, the following additional specifications were added:

a. Within the constraints of the sponsor's specifications concerning cost and size, every effort should be made to keep the MAPU as compatible with Autodin operation as possible.

b. The MAPU should be examined during all phases of redesign for the purpose of making improvements where advances in state-of-the-art and sponsor constraints allow.

c. In conjunction with b above, the operation of the MAPU should be simplified and automated wherever possible in order to reduce human interface and subsequent possibility of error.

The goals of this thesis in order of their importance are: the production of a working MAPU redesign on paper; a working breadboard design which will interface with MAPU components on loan from the Air Force Depot at McClellan AFB,

and finally a successful operational trial period on board the AN/TGC-26 at Wellesley during a 48-hour live traffic exercise. Various design stages are certified at the Air Force Institute of Technology, using the Institute's resident Digital Logic Simulator (DLS) program and CYBER computer system.

Chapter II of this thesis goes through a general description of MAPU logic operation. Chapter III focuses in more detail on the code recognition circuitry of the MAPU for a clear understanding of how it does its job, and for an insight as to how it can be modified to process JANAP 128 format. Chapter IV deals with the use of the Digital Logic Simulator in modeling each stage of circuit modifications. Chapter V reports on the construction of a working piece of hardware and subsequent testing at Wellesley, Massachusetts. Finally, Chapter VI contains recommendations for improvements beyond the scope of this report.

II. OPERATION OF THE MULTIPLE ADDRESS PROCESSOR

MAPU Operation

Before attempting to redesign the logic of the MAPU, a thorough understanding of MAPU operation in its existing condition using ACP 127 format is required. Chapter I gave a general description of the steps that an operator would take to process a message tape. This chapter will go through a general description of the logic operation of the entire MAPU with the aid of functional block diagrams. All references will be made to the block diagram in Fig. 6, and to the timing diagram in Appendix B, Fig. 30. The reader may wish to refer to Appendix C, Figs. 31 through 34 and Appendix D, Fig. 35 for detailed schematic diagrams of the various sections discussed in this chapter.

Block Diagram Analysis

Fig. 6 provides a detailed block diagram of the MAPU, including the primary control assembly and the power supply and tape reader assembly. The secondary control assembly is not shown as it duplicates the functions of the primary control assembly for an additional six reperforators, giving a total of 12 slave reperforators.

The following block diagram components are shown in Fig. 6:

a. Paper Tape Reader and tape reader wetting circuit which read and convert five bit Baudot code punches

in the tape to logic level signals for the MAPU logic circuitry.

b. TD driver logic and TD drive circuits which generate the TD DRIVE COIL CURRENT signal to step the paper tape in the reader.

c. Item select logic which routes an address to the correct slave reperforator by depressing the ITEM SELECTION pushbutton.

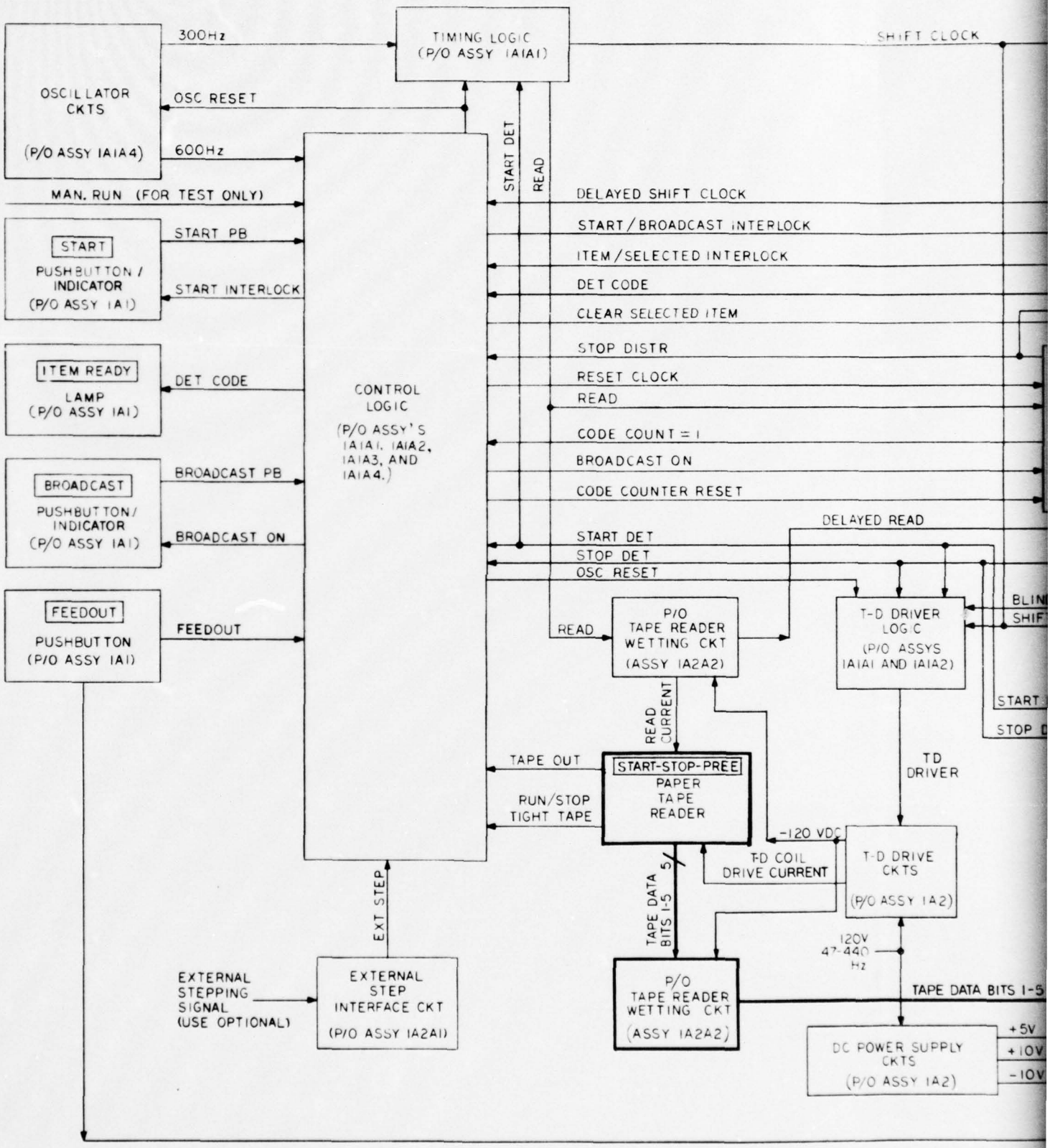
d. Oscillator circuits, timing logic, and a 0.75 bit interval delay circuit which provide the MAPU with internal timing signals and determine the transmission rate (baud).

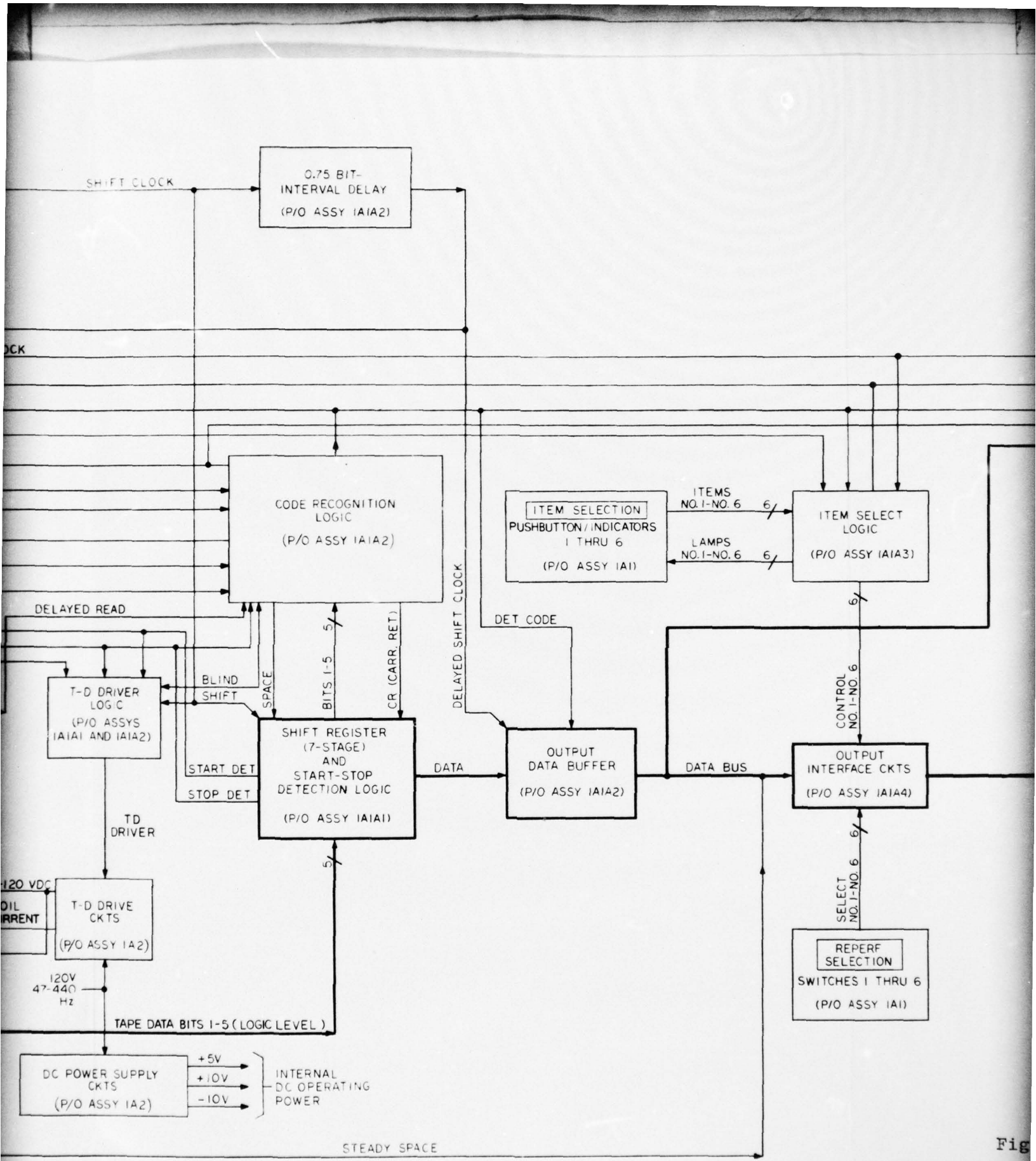
e. Control logic which converts operator commands into logic signals to sequentially step the MAPU through the various modes.

f. A seven-stage shift register and start-stop detection logic to convert five bit parallel input into standard seven bit start-stop output characters for serial output transmission.

g. Code recognition logic which detects the presence of a preselected flag character, sets the flip-flop associated with that character, stops the MAPU tape reader, and locks the output data buffer in a mark hold condition to inhibit transmission of the detected character to the output interface circuits.

h. Output Data Buffer which transfers serial data from the shift register to the output interface circuits.


OSCILLATOR CKTS
(P/O ASSY 1A1A4)
300Hz
OSC RESET
600Hz
TIMING LOGIC
(P/O ASSY 1A1A1)
SHIFT CLOCK
START DET
READ
MAN. RUN (FOR TEST ONLY)
START
PUSHBUTTON / INDICATOR
(P/O ASSY 1A1)
START PB
START INTERLOCK
ITEM READY
LAMP
(P/O ASSY 1A1)
DET CODE
BROADCAST
PUSHBUTTON/ INDICATOR
(P/O ASSY 1A1)
BROADCAST PB
BROADCAST ON
FEEDOUT
PUSHBUTTON
(P/O ASSY 1A1)
FEEDOUT
CONTROL LOGIC
(P/O ASSY'S 1A1A1, 1A1A2, 1A1A3, AND 1A1A4.)
DELAYED SHIFT CLOCK
START/BROADCAST INTERLOCK
ITEM/SELECTED INTERLOCK
DET CODE
CLEAR SELECTED ITEM
STOP DISTR
RESET CLOCK
READ
CODE COUNT = 1
BROADCAST ON
CODE COUNTER RESET
DELAYED READ
START DET
STOP DET
OSC RESET
READ
P/O TAPE READER WETTING CKT
(ASSY 1A2A2)
T-D DRIVER LOGIC
(P/O ASSYS 1A1A1 AND 1A1A2)
READ CURRENT
TAPE OUT
RUN/STOP TIGHT TAPE
START-STOP-PREE
PAPER TAPE READER
TD DRIVER
-120 VDC
T-D COIL DRIVE CURRENT
T-D DRIVE CKTS
(P/O ASSY 1A2)
TAPE DATA BITS 1-5
5
EXT STEP
120V 47-440 Hz
EXTERNAL STEPPING SIGNAL (USE OPTIONAL)
EXTERNAL STEP INTERFACE CKT
(P/O ASSY 1A2A1)
P/O TAPE READER WETTING CKT
(ASSY 1A2A2)
TAPE DATA BITS 1-5
DC POWER SUPPLY CKTS
(P/O ASSY 1A2)
+5V
+10V
-10V


SHIFT CLOCK
0.75 BIT-INTERVAL DELAY (P/O ASSY 1A1A2)
CODE RECOGNITION LOGIC (P/O ASSY 1A1A2)
ITEM SELECTION PUSHBUTTON/INDICATORS 1 THRU 6 (P/O ASSY 1A1)
ITEMS NO.1-NO.6
LAMPS NO.1-NO.6
ITEM SELECT LOGIC (P/O ASSY 1A1A3)
DELAYED READ
DET CODE
DELAYED SHIFT CLOCK
BLIND SHIFT
SPACE
BITS 1-5
CR (CARR. RET.)
CONTROL NO.1-NO.6
T-D DRIVER LOGIC (P/O ASSYS 1A1A1 AND 1A1A2)
SHIFT REGISTER (7-STAGE) AND START-STOP DETECTION LOGIC (P/O ASSY 1A1A1)
START DET
STOP DET
DATA
OUTPUT DATA BUFFER (P/O ASSY 1A1A2)
DATA BUS
OUTPUT INTERFACE CKTS (P/O ASSY 1A1A4)
TD DRIVER
-120 VDC
OIL
RRENT
T-D DRIVE CKTS (P/O ASSY 1A2)
SELECT NO.1-NO.6
REPERF SELECTION SWITCHES 1 THRU 6 (P/O ASSY 1A1)
120V 47-440 Hz
TAPE DATA BITS 1-5 (LOGIC LEVEL)
DC POWER SUPPLY CKTS (P/O ASSY 1A2)
+5V
+10V
-10V
INTERNAL DC OPERATING POWER
STEADY SPACE

Fig

8

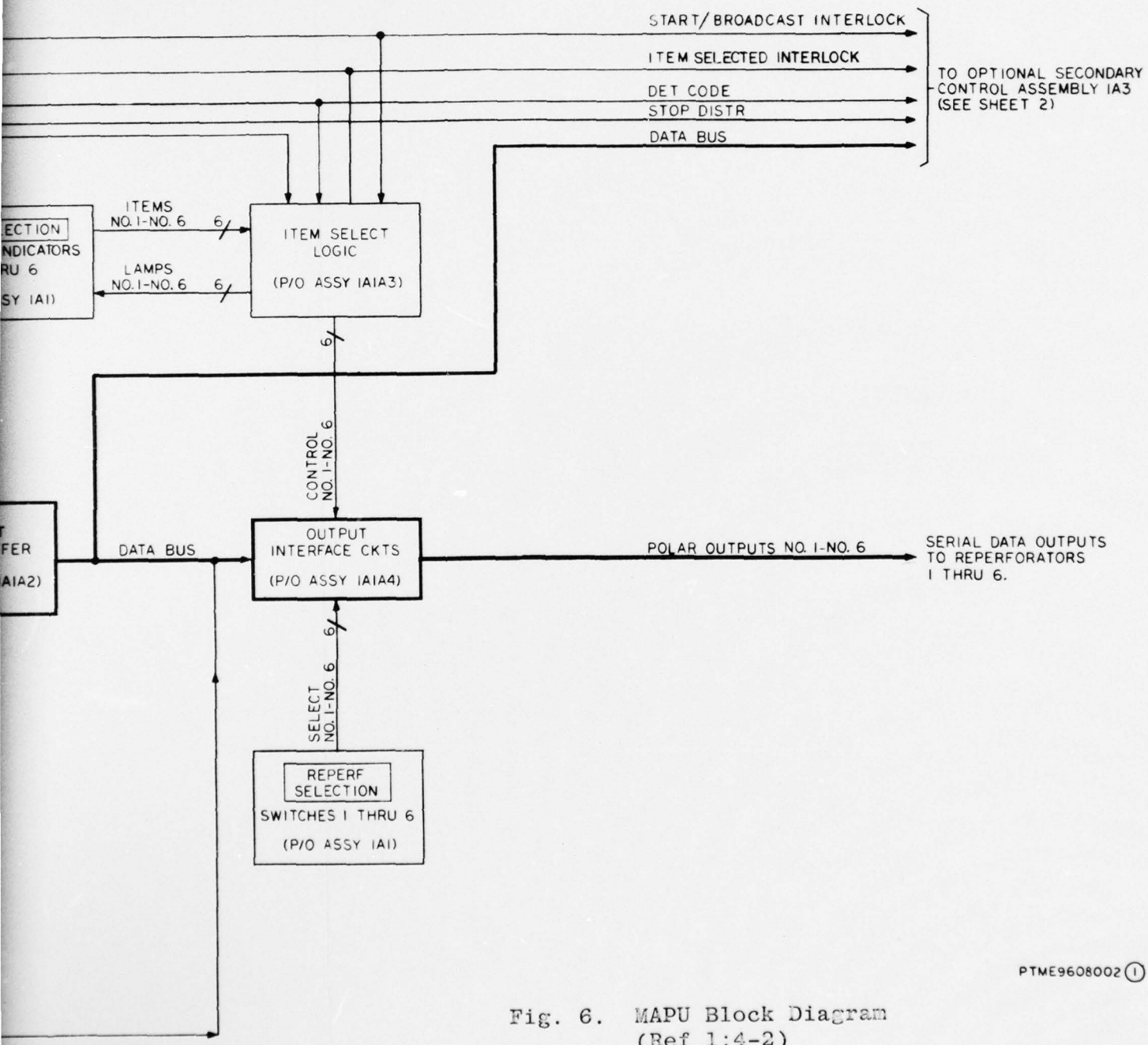


Fig. 6. MAPU Block Diagram
(Ref 1:4-2)

3

The data buffer actually forms the eighth stage of the shift register described in f above; however, the data buffer is clocked by the DELAYED SHIFT CLOCK signal which is a 0.75 bit interval delay later than SHIFT CLOCK which clocks the first seven stages. This delay allows time for the code recognition logic to detect a flag character and inhibit its transmission to the DATA BUS.

i. Output interface circuits which convert serial data on the DATA BUS to six identical polar level outputs and gate these outputs to the six reperforators under the control of item select logic (Ref 1:4-8).

Basic Timing. The oscillator circuits of Fig. 6 generate a 300 Hz output which is divided down by the timing logic to provide a baud rate (strap selected as 37.5, 50, or 75 baud) SHIFT CLOCK signal to the shift register. Each negative-going transition of this signal shifts the contents of the register one stage right. SHIFT CLOCK is also delayed 0.75 bit interval to provide DELAYED SHIFT CLOCK to the output data buffer where each negative-going transition shifts one bit into the buffer. These signals determine the MAPU transmission rate and allow for the required delay to inhibit transmission of a flag character to the reperforators.

The READ pulse is generated by the timing logic at the mid-point in each start-space bit interval under control of the START DET signal from start-stop detection logic. The READ pulse is supplied to the tape reader wetting circuit

where it is converted into the READ CURRENT pulse used to read the paper tape in the reader. The tape wetting circuit delays the READ pulse by 20 microseconds to provide the DELAYED READ signal for code recognition logic. The DELAYED READ signal allows gating of the contents of the shift register into the code recognition logic only after the READ pulse has completed the transfer of information from the paper tape to the register. The READ signal is also supplied to the code recognition logic to transfer the stored flag character back into the shift register when the MAPU is restarted for routing indicator transmission (Ref 1:4-9).

The oscillator circuits and timing logic are controlled by the OSC RESET signal. When OSC RESET is set active by the control logic, all timing and oscillator circuits are stopped and reset, thereby stopping the MAPU. The control logic does not allow the active state while a character is being transmitted, therefore the MAPU can only halt after the last bit (stop-mark) of a character has been transmitted.

Tape Reader Control. TD driver logic combines the START DET and STOP DET signals from the start-stop detection logic with the SHIFT CLOCK signal to produce a TD DRIVER signal which is active during the stop-mark interval and the first half of the following start-space interval except when disabled by the BLIND signal from code recognition logic. The TD DRIVER signal in its active state produces a TD COIL DRIVER CURRENT to the tape reader drive coil. This signal raises the bit-sensing pins to contact the tape and sense character

punches. While the pins are in the up position, the READ signal produces the READ CURRENT pulse which parallel transfers the data from those pins in contact with the tape (no punch holes) to the shift register. This action occurs coincident with the mid-point of the start-space bit interval. When the TD DRIVER becomes inactive, the sensing pins are withdrawn and the tape reader steps the tape to the next character. This cycle is repeated at the beginning of each character interval (seven bit intervals), with the exception of the case when the BLIND signal is active. The BLIND signal is activated each time the code recognition logic detects and stores a flag character. BLIND remains active during the first character interval following the restart of the MAPU to enable the re-insertion and transmission of the stored character. The BLIND signal is also used to disable the code recognition circuitry during the flag character re-insertion to prevent detection. Code recognition logic resets BLIND to its inactive state when the STOP DET signal becomes active at the end of the flag character interval. At this time the tape reader resumes normal read and step cycles.

Data Transfer Control. Fig. 7 shows a listing of Baudot code characters and their associated seven bit serial representation. Whenever the MAPU is stopped, the shift register is in the stop-mark condition, in which the rightmost stage is storing a stop-mark (1-state), and the remaining six stages store start-spaces (0-state). This condition activates the STOP DET signal of the start-stop detection logic. When the

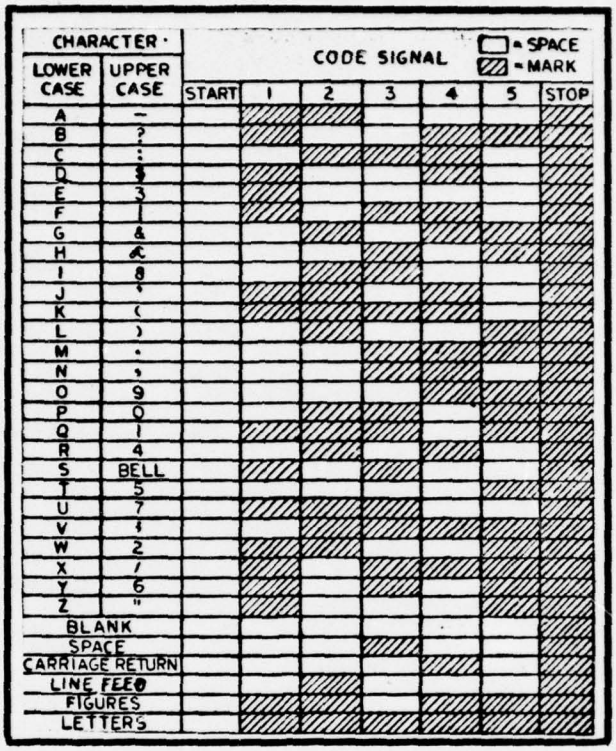


Fig. 7. Standard Seven Unit (Five Data Bit) Start-Stop Baudot Code

MAPU is started, by either depressing the START, ITEM SELECT, or BROADCAST pushbuttons, the control logic clears the OSC RESET signal, activating the TD driver and timing logic.

The first SHIFT CLOCK pulse from the timing logic shifts a start-space into the rightmost stage of the shift register and inserts a stop-mark into the leftmost stage. The shift register is now conditioned to receive five data bits in the middle five stages at the first READ pulse. The start-stop detection logic now recognizes the start-space condition and deactivates the STOP DET signal and activates the START DET signal. The active START DET signal enables the timing logic to generate the READ signal, which appears approximately in

the mid-point of the start-space interval. The READ signal parallel transfers the five data bits into the shift register and the register now contains the seven bits of the standard start-stop code. The start-space bit is shifted into the output data buffer by the DELAYED SHIFT CLOCK signal 0.75 bit interval after it is shifted into the register output stage (rightmost) by the first SHIFT CLOCK signal. The start-space therefore appears on the DATA BUS 0.25 bit interval after the READ pulse has loaded the register. This 0.25 bit interval delay allows time for the code recognition logic to detect a flag character and lock the output data buffer in the mark condition (1-state) by means of the DET CODE signal applied to the preset of the output data buffer flip-flop.

The next five SHIFT CLOCK pulses transfer the five data bits into the shift register output stage where they are subsequently shifted into the output data buffer by the DELAYED SHIFT CLOCK pulses. The seventh SHIFT CLOCK signal shifts the stop-mark bit which was previously inserted into the leftmost stage into the register output stage where the start-stop detection logic senses the stop-mark condition (the leftmost stage has been functioning as a source of start-spaces to left fill the first six stages with spaces (1-state) as each SHIFT CLOCK pulse occurs) and activates STOP DET. If the character in the cycle discussed above is a flag character, then the MAPU would be commanded to halt at the end of the character interval. If the character is not a flag, then the cycle repeats with the next SHIFT CLOCK pulse. Appendix C,

Fig. 31 shows a detailed schematic diagram of the character distribution circuits, including the seven-stage shift register (Ref 1:4-9, 6-1).

Message Heading Transmission Cycle

At this time it would be useful to follow the MAPU through its operations as a message tape is read up to the first flag character. After the message tape is inserted in the tape reader and the tape reader signal is set to the RUN state, the following events occur upon depressing the START push-button:

a. The START INTERLOCK signal is activated, illuminating the START pushbutton.

b. The START/BROADCAST INTERLOCK signal is activated, allowing the item select logic to activate all CONTROL signals to output interface circuits. This allows data to be passed to all reperforators previously switch-selected by the operator.

c. The CODE COUNTER RESET pulse is activated to reset the code recognition logic to its initial state (code count = 3). This allows the code recognition logic to detect a CR character as the first flag in the end-of-message-heading code. See Fig. 7 for CR code.

d. The OSC RESET signal is set to its inactive state. This allows the oscillator and timing circuits to run and start the data transfer cycle discussed previously.

Just after each character is read into the shift register by the READ signal, and before it is shifted out by the SHIFT CLOCK signal, the code recognition logic tests the data for a CR character. If the CR is present, the code count indexes to 2 and the data transfer continues. Code count = 2 conditions the code recognition logic to detect a LF character as the second character in the end-of-message-heading code. When the LF character is detected by the code recognition logic, the code count is indexed to 1 and the logic is now conditioned to detect either a CR or a SP as the final flag in the end-of-message-heading code. It should be pointed out that any number of non-flag characters may be interspersed between flag characters.

When the code recognition logic detects either a SP or a CR, the following events occur:

a. The code recognition logic generates a STOP DISTR signal to command the MAPU to halt.

b. The appropriate character flip-flop is set.

c. The DET CODE signal is activated.

These events occur coincident with the DELAYED READ signal, approximately 20 microseconds after the READ signal transfers the character into the shift register. The STOP DISTR command to the MAPU activates the OSC RESET signal which will stop and reset the oscillator and timing logic after the character (SP or CR) has been shifted out of the register and the start-stop detection logic senses the stop-mark condition and activates STOP DET. The active DET CODE signal has the

following effects:

a. Locks the output data buffer in the stop-mark condition during the period when the SHIFT CLOCK and DELAYED SHIFT CLOCK signal are shifting the flag character out of the register. The output data register therefore acts as a bit sink as the character is shifted out and lost.

b. Deactivates the START INTERLOCK and START/BROADCAST INTERLOCK signals to extinguish the START pushbutton, and inhibit output to reperforators.

c. Activates the BLIND signal in the code recognition logic in order to inhibit the tape read and step cycle for one character interval following the restart of the MAPU. This inhibition allows the reinsertion of the stored flag character. The BLIND signal also disables the code recognition logic to prevent detection of the reinserted flag character (Ref 1:4-16).

The code recognition logic will remain in a code count = 1 state in order to detect CRs or SPs which are the end-of-routing-indicator codes. The code recognition logic remains in this state until the BROADCAST pushbutton is depressed. The BROADCAST ON signal is then activated which resets and holds the code recognition logic in the code count = 3 condition for the remainder of the message tape.

III. CODE RECOGNITION LOGIC

Signal Convention

In Chapter II the operation of the MAPU as a whole was examined. Now a more detailed discussion of the code recognition portion of the MAPU is in order. MAPU signals are identified by functional names that describe conditions under which the signals are active, or indicate functions performed by the active state of the signals. A bar over the signal name indicates that the active state of the signal is a logical low level. Absence of the bar indicates that the active state of the signal is a logical high. For example, signal DET CODE is the functional name given to that signal generated when a code is detected. The signal name indicates that it is a logical high state when the condition code detected exists. Likewise, $\overline{\text{DET CODE}}$ is a logical low level signal when the condition code detected exists. The exception to this convention is the block diagram of Fig. 6. Signal names appear without the "not" designation in order to simplify the diagram.

The DLS modeling program limits signal names to five alpha-numeric characters, therefore the signal names appearing in the DLS simulations in Chapter IV will be truncated to five letters. These signals will, however, correspond to the signals shown on all electrical schematics (Ref 1:4-1). For example, again using the signal DET CODE, the DLS simulations will truncate the signal name to DETCO.

Code Recognition Logic Flow

Fig. 8 shows a flow chart presentation of the code recognition logic functions. The act of applying power to the MAPU activates an initializing circuit which insures that the code counter is preset and the shift register is cleared (Ref 1:4-27). The START signal is activated by depressing the start pushbutton, and the code count = 3 condition of the code recognition logic allows it to test each character for a CR as the MAPU cycles through the shift/read process described in Chapter II. When a CR is detected, the code counter is decremented by one and the cycle continues. The code recognition logic is now conditioned to detect a LF character. Upon detection of this character, the code counter is decremented by one and code count = 1. The code recognition logic is now conditioned to detect either a CR or a SP character. When either of these two characters is detected, the code recognition logic sets the character flip-flop, deactivates the $\overline{\text{BLIND}}$ signal, and halts the MAPU. The message heading has now been completely transmitted. The operator may now elect to segregate routing indicators by depressing ITEM SELECT buttons corresponding to the desired slave reperforator, or he may transmit the remainder of the message to all reperforators by depressing the broadcast button. If ITEM SELECT is activated, the code recognition logic remains at code count = 1, and the logic continues to test for a CR or a SP character as the flag for the end of each routing indicator. If the operator elects to activate BROADCAST, the code counter

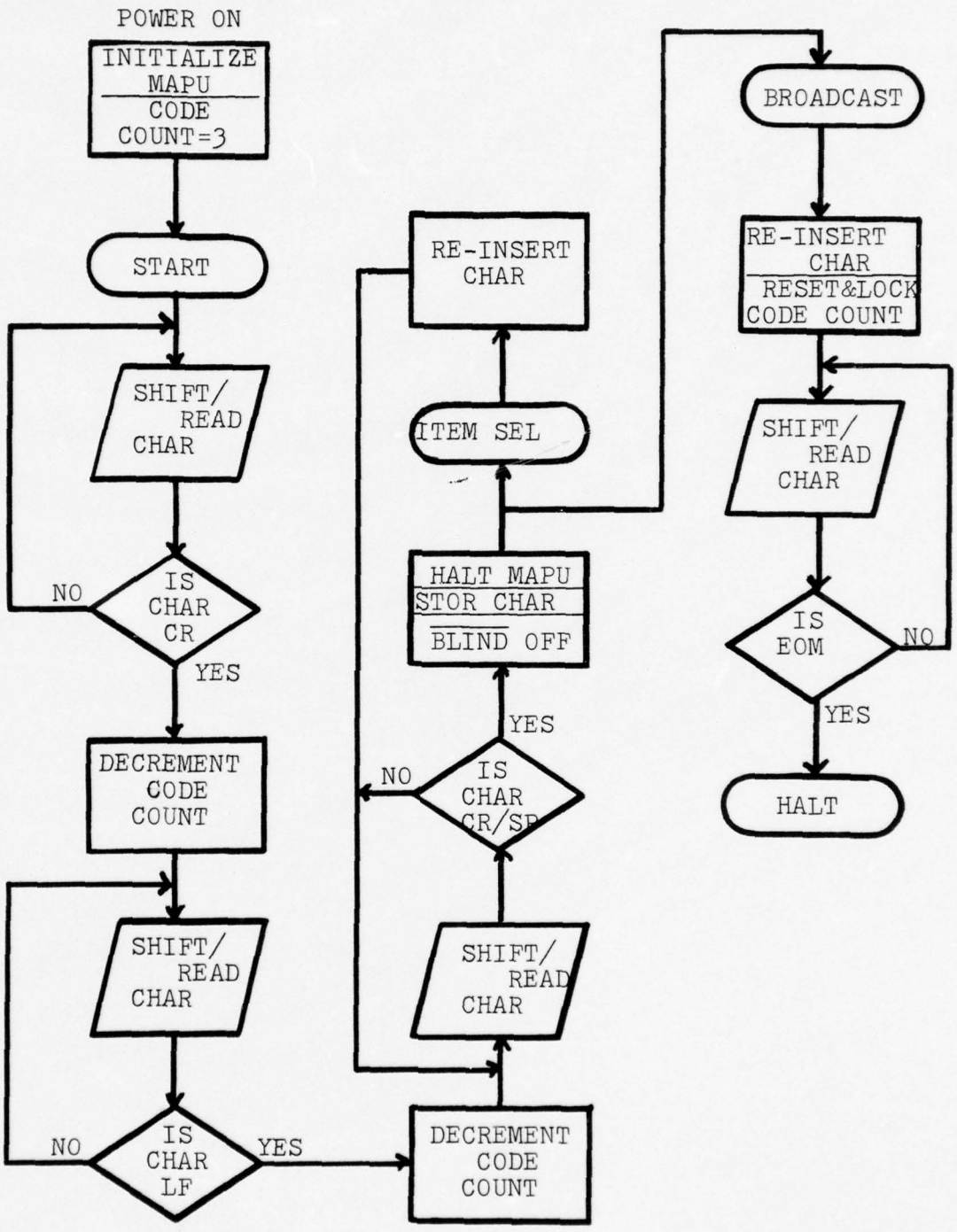


Fig. 8. Code Recognition Logic Flow Diagram

is reset to count = 3 and locked in this condition to preclude any further detection. The shift/read cycle will continue uninterrupted until the end-of-message (EOM) signal is detected. The MAPU is now conditioned to accept the next message tape.

Code Recognition Logic Circuits

The actual schematic diagram of the code recognition logic appears in Appendix C, Fig. 32. A check of the logic device numbers reveals that the MAPU was constructed using Diode-Transistor Logic (DTL). The significance of this fact will be discussed in Chapter V. Because some logic packages contain many discrete gates, a convention will be used in the following circuit analysis. A discrete gate will be referred to by the integrated circuit (IC) package number of which that gate is a part, followed by the package pin number which defines that gate's output. Therefore gate IC1-1 indicates that gate contained in IC1 whose output appears on pin 1 (Ref 1:4-23).

When the START pushbutton is depressed, the START signal is passed through a differentiator circuit, and the resulting positive-going pulse is applied to gate IC6-8 which is enabled by an active OSC RESET signal. $\overline{\text{CODE COUNTER RESET}}$ is the resulting negative-going spike which is passed through gates IC10-12 and IC3-3 and inverters IC7-4 and IC7-12 to the preset pins of IC8-6, IC8-8, and IC5-8. The flip-flops are preset, and the Q output of IC5-8 enables gate IC6-11 to pass the $\overline{\text{STOP DET}}$ signal which was shaped as a negative-going spike by a differentiator and inverted by IC7-2. This spike sets the

BLIND flip-flop, IC6-6, and resets IC3-11 and IC1-6, the CR and SP flip-flops respectively. The Q outputs of IC8-6 and IC8-8, along with the high $\overline{\text{BLIND}}$ signal, enable gate IC4-8 to detect a CR character through the programmed network formed by inverters IC2-3, IC2-2, IC2-8, IC2-10, and IC2-12.

When IC4-8 is enabled by the appearance of a CR in the message heading, the $\overline{\text{DELAYED READ}}$ pulse toggles flip-flop IC8-6 via inverter IC7-8. IC4-8 is now disabled by the low Q output of IC8-6, and IC4-6 is enabled by the combination of the high $\overline{Q}$ of IC8-6 and Q of IC8-8. IC4-6 is programmed by the inverter network to detect a LF character. When this character occurs, the DELAYED READ pulse once again toggles IC8-6 and the resulting high Q output of IC8-6 and low Q output of IC8-8 disables IC4-6 and enables gates IC11-8 and IC11-6. IC11-8 is programmed to detect a SP character and IC11-6 is programmed to detect a CR. The code recognition logic is now conditioned to halt the MAPU upon detection of either of these characters.

When a SP or a CR occurs, either IC11-8 or IC11-6 passes the DELAYED READ pulse and the following events occur:

a. The DELAYED READ pulse toggles flip-flop IC5-8, whose Q and $\overline{Q}$ outputs provide DET CODE and $\overline{\text{DET CODE}}$ respectively. $\overline{\text{DET CODE}}$ locks the output data buffer in the stop-mark condition and resets the BLIND flip-flop IC6-6, while DET CODE enables the ITEM SELECT logic. The active $\overline{\text{BLIND}}$ signal inhibits the TD driver gate, IC9-8, and disables gates IC11-8 and IC11-6.

b. The DELAYED READ pulse is passed through IC9-11 and inverter IC3-6 to form the $\overline{\text{STOP DIST}}$ signal to halt the tape reader.

c. Either SP flip-flop IC1-6 or CR flip-flop IC3-11 is set, enabling gate IC3-8 or IC9-6 respectively to pass the $\overline{\text{READ}}$ signal the next time the MAPU is started. The $\overline{\text{READ}}$ pulse will reload either a SP or a CR character into the shift register of the character distribution logic (See Appendix D for bit bus connections between circuit cards) (Ref 1:4-41).

At this time the MAPU is halted at the end-of-message-heading information. When the operator selects an ITEM SELECT pushbutton, the resulting $\overline{\text{READ}}$ pulse reloads the SP or CR character by means of IC3-8 or IC9-6. The RESET CLOCK signal which occurs at the start of each start-space passes through inverter IC7-6, gate IC3-3, and inverter IC7-12 to the preset pin of flip-flop IC5-8. The Q output of IC5-8 goes high and enables gate IC6-11 to pass the $\overline{\text{STOP DET}}$ signal to set the BLIND flip-flop, IC6-6. This action removes the disabling $\overline{\text{BLIND}}$ signal from gates IC11-8 and IC11-6, allowing them to detect the next SP or CR character that occurs in the routing indicator portion of the message.

The process described above is repeated each time that a SP or CR character is detected and an ITEM SELECT button is depressed. When the MAPU halts at the end of the last routing indicator, the operator depresses the BROADCAST pushbutton. The high level signal from the BROADCAST pushbutton

is shaped into a positive-going pulse by a differentiator and passed by gate IC10-8, which is enabled by the high $\overline{Q}$ of IC8-8. The resulting $\overline{\text{BROADCAST}}$ signal sets the BROADCAST flip-flop IC5-8 of the control panel circuit assembly (See Appendix C, Fig. 33). The BROADCAST flip-flop returns a $\overline{\text{BROADCAST ON}}$ signal which is passed by gates IC10-6 and IC10-12 and inverter IC7-4 to the preset pins of flip-flops IC8-6 and IC8-8, locking them in a set condition for the duration of the remainder of the message. With IC8-6 and IC8-8 held in the set condition, gates IC11-8 and IC11-6 are disabled and no further character detection can take place (Ref 1:4-43).

IV. INITIAL MODIFICATION DESIGN AND SOFTWARE TESTING

JANAP 128 Conversion

Chapter III examined the operation of the original code recognition logic in detail. This chapter discusses the required modifications of the logic circuitry to perform its new task. New flag characters are selected, and the design changes are carried out in three main stages. Each stage is an improvement over the previous one as the design gradually realizes all its functions. The Digital Logic Simulator (DLS), a relatively new computer tool, is used extensively to verify the operation of logic designs.

Flag Character Selection. The first task in realizing a new design is the proper selection of flag characters to trigger MAPU halts and mode changes. Fig. 5 of Chapter I shows a typical message formatted in JANAP 128. The first halt must occur immediately before the routing indicator RUEKASA. Immediately preceding this routing indicator are two "dash" characters. At first glance these characters seem suitable to trigger the code recognition circuitry, but a check of Fig. 7 of Chapter II indicates that the "dash" character shares its five bit code with the letter "A" in lower case. Since the letter "A" is a commonly occurring character, the double "dash" combination is unreliable. There is, however, a character which occurs between the second "dash" and the beginning of the routing indicator. This character is the

LT character, one of six machine operation codes which are single-valued, that is, their baudot code is unique to them. Re-examining the first line of the message in Fig. 5, the following unique combination is found: a FG character occurs between the last letter of the originating station routing indicator and that station's serial number; a LT character occurs between the "dash" and the first letter of the classification code; a FG character occurs between the last letter of the classification code and the double "dash" combination; finally, a LT character occurs between the double "dash" and the first letter of the first routing indicator. This combination of FG, LT, FG, LT is fixed by the nature of JANAP 128 format and will always occur in that order. Therefore, this is the combination of characters chosen to index the code recognition logic to its first halt.

The next step in the design is to find the character or characters which act as routing indicator halt flags. Again referring to Fig. 5, the following characters are found to separate routing indicators: a SP character, and a combination of CR, CR, LF. The SP character is an obvious choice, but the three-character combination presents some problems. The CRLF combination signals an end-of-line situation in teletype messages. Since the position of the CRLF combination in the master tape has no relation to the end-of-line situation in the slave tape, a situation where the CRLF following the fourth routing indicator on the master tape is used to separate the first routing indicator on the slave tape might

occur. A solution to the problem is to store and re-insert a LT character everytime a CRLF combination occurs and the given ITEM SELECT button is depressed for the first time. If the ITEM SELECT button has already been selected one or more times, then a SP character would be substituted for the CRLF. This scheme would insure that every slave tape begins with a re-inserted LT character to position the reperforator in the lower case. This same scheme would apply in the occurrence of a SP separator, and a first-time selection of an ITEM SELECT button.

In the event of the occurrence of the CRLF combination, the data buffer must be locked on the first CR. The MAPU is allowed to continue to step the tape until the LF is detected. Such split operation of the MAPU is not found in the original logic and therefore must be designed in. In the original circuit the signal $\overline{\text{DETCODE}}$ performed the functions of locking the data buffer and halting the tape reader. A new signal is now required which can act in advance of the $\overline{\text{DETCODE}}$ signal. This signal will be referred to as $\overline{\text{LOCK}}$ and will require a memory (flip-flop) to determine if a CR has occurred previously.

The final step in the design process is solving the problem of insuring that a period terminates every slave tape. This period is a required part of the JANAP 128 format and is critical in the proper processing of the message by AUTODIN (Ref 3). This problem is easily solved when it is noted that the period is preceded by a FG character. The appearance of the FG character is used to index the counter out of the item

select mode, halt the MAPU, and store the FG character. Depressing the BROADCAST button re-inserts the FG character, and everything following is simultaneously transmitted to all activated reperforators. This action automatically places the period in the correct location on each slave tape.

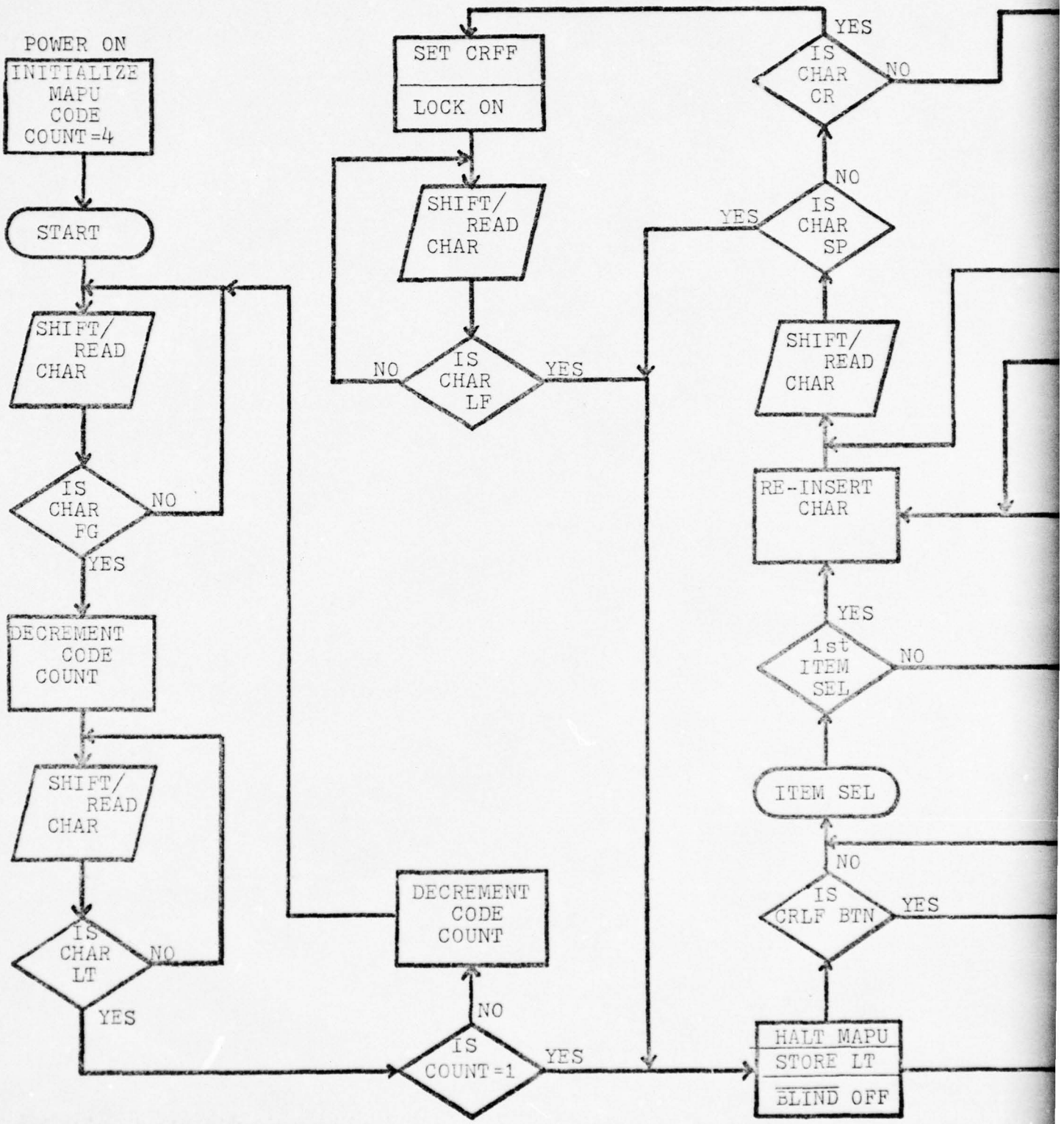
In order to add an additional convenience feature not found in the original design to the MAPU, the appearance of the FG character is used to automatically index the MAPU into the broadcast mode. This action eliminates the manual selection of the broadcast mode by the operator. Broadcast mode can still be initiated at any time previous to the occurrence of the last FG character (Ref 3).

Re-Insertion of the CRLF

The problem of removing the CRLF combination when it is not wanted has been discussed. There remains the problem of inserting the CRLF combination where it is needed. In JANAP 128 format there is character space on the first line for no more than four routing indicators. On each succeeding line there is space for a maximum of eight. When the number of routing indicators on the first line of each slave tape reaches four, a CRLF combination must be manually or automatically injected before the next routing indicator. The manual injection procedure is chosen because of the physical limitations of space to contain the circuitry. Six ITEM SELECT button monitors are required to keep track of the number of times each button is depressed. Chapter VI will discuss the circuitry for automatic injection of the CRLF.

Modified Code Recognition Logic Flow

Fig. 8 of Chapter III shows the logic flow diagram for the original MAPU code recogntion logic. Fig. 9 demonstrates how this flow is modified to recognize the flags chosen for JANAP 128. When power is applied, the MAPU is initialized and the code recognition counter is set to count = 4. The MAPU cycles through the shift/read process until the first FG character is detected. The code counter is decremented to count = 3, and the cycle continues until the LT character is detected. Code count now equals 2 and the logic is conditioned to detect the next FG character. When the FG character occurs, the code counter is decremented to count = 1, and the logic is conditioned to detect the next LT character. When the LT character occurs, the code recognition logic sets the LT flip-flop, halts the MAPU, and turns off the $\overline{\text{BLIND}}$ signal. The operator may now elect to segregate routing indicators by depressing ITEM SELECT buttons corresponding to the desired slave reperforator, or he may transmit the remainder of the message to all activated reperforators by depressing the BROADCAST button. If ITEM SELECT is activated, the code recognition logic re-inserts the LT character, turns on the $\overline{\text{BLIND}}$ signal, and starts the tape reader in the shift/read cycle. The code recognition logic is now conditioned to detect a SP, a CR, or a FG character. If a SP is detected, the MAPU halts, $\overline{\text{BLIND}}$ is turned off, and the LT flip-flop is set. If the code recognition logic detects a CR, the CR flip-flop is set and the $\overline{\text{LOCK}}$ signal is activated. The data buffer is locked


POWER ON
INITIALIZE MAPU CODE COUNT=4
START
SHIFT/ READ CHAR
IS CHAR FG
NO
YES
DECREMENT CODE COUNT
SHIFT/ READ CHAR
IS CHAR LT
NO
YES
SET CRFF
LOCK ON
SHIFT/ READ CHAR
IS CHAR LF
NO
YES
DECREMENT CODE COUNT
NO
IS COUNT=1
YES
IS CHAR CR
YES
NO
IS CHAR SP
NO
YES
SHIFT/ READ CHAR
RE-INSERT CHAR
1st ITEM SEL
YES
NO
ITEM SEL
IS CRLF BTN
NO
YES
HALT MAPU
STORE LT
BLIND OFF

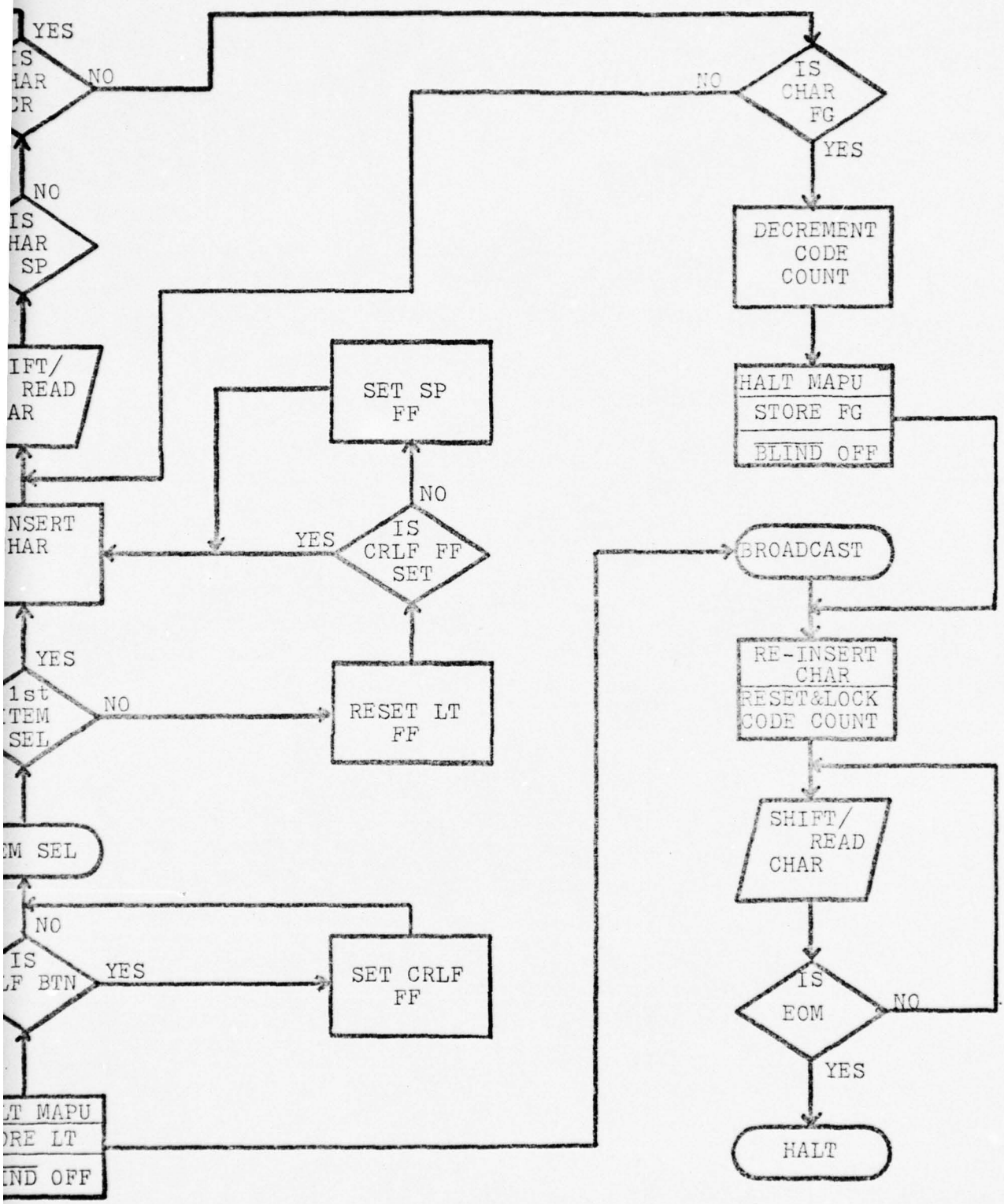


Fig. 9. Modified Code Recognition Logic Flow Diagram

in the mark-hold condition and the logic is conditioned to detect a LF. When the LF character occurs, the MAPU halts, the $\overline{\text{BLIND}}$ signal is turned off, and the LT flip-flop is set.

If the next ITEM SELECT button depressed is activated for the first time, the LT character is re-inserted. If the button is depressed for the second or subsequent time, the LT flip-flop is cleared and the SP flip-flop is set, followed immediately by insertion of the SP character. If the CRLF button is depressed prior to the ITEM SELECT button, the LT flip-flop is cleared, and the CRLF injector is unlocked.

If the code recognition logic detects a FG character, the code counter indexes once more and initiates the $\overline{\text{BROADCAST}}$ signal which is routed to the control panel logic. The MAPU halts momentarily to store the FG character and turn off the $\overline{\text{BLIND}}$ signal. The MAPU restarts almost instantly as the returned $\overline{\text{BROADCAST ON}}$ signal (See Appendix C, Fig. 32) from the control panel logic initializes and locks the code counter in the count = 4 condition, re-inserts the FG character, and turns on the $\overline{\text{BLIND}}$ signal (Ref 1:6-3). The MAPU now continues the shift/read cycle uninterrupted until the EOM signal is detected. Selecting the BROADCAST button at any time prior to the occurrence of the FG character would start the normal sequence of events for broadcast mode described in Chapter III.

Isolation of Circuitry to be Modified

In order to remain within the constraints of cost and the need to retain the capability of processing ACP 127 format, the modification is limited to a single circuit card in the

1A1 Assembly (See Appendix D, Fig. 35). This card is the code recognition card, Assembly A2. The diagram in Fig. 10 is an expansion of the code recognition block in the block diagram of Fig. 6. The logic circuitry is viewed as an empty box initially. The design process follows a step-by-step procedure in which circuit elements are added such that, given only the original available inputs, all the desired outputs are generated. The signal SELC1 marked with a single asterisk is the only signal required in the modification that is not already present at the terminals of the A2 card. This signal will be explained in detail later. The $\overline{\text{LOCK}}$ signal, shown internal to the block, has already been discussed. It is generated and utilized within the confines of the A2 circuit card. The signals LT, FG, and LF are new signals, marked with double asterisks, while the CR and SP signals are part of the existing circuit.

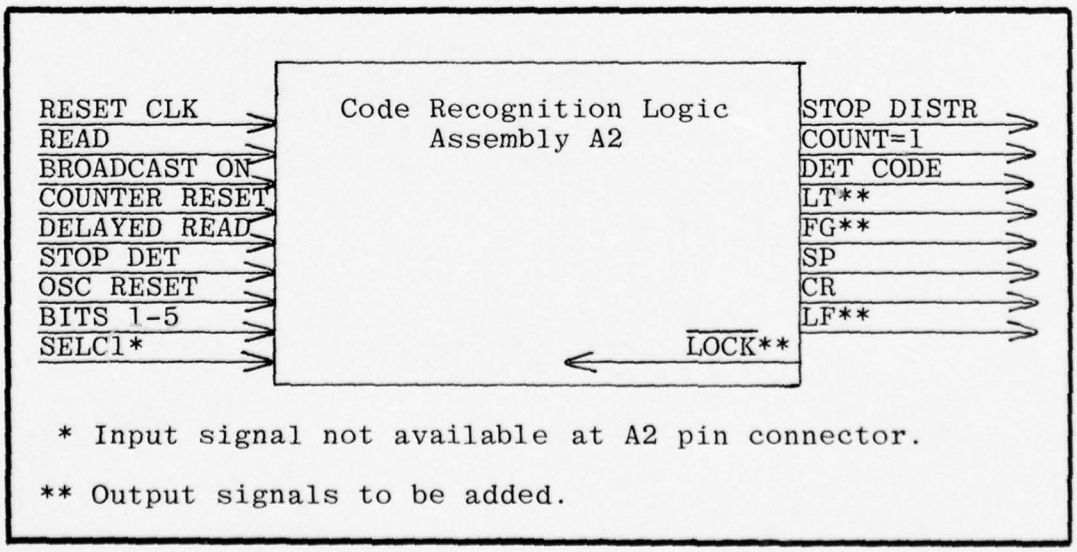


Fig. 10. Modified Code Recognition Logic Block Diagram

The confines of the problem are now defined, and the specific goals have been enumerated. Before attempting hardware design and construction, the redesigned logic operation is verified using computer simulation.

Computer Modeling and Simulation

The Digital Logic Simulator is a computer program which can simulate digital logic circuits. The program is written in Fortran IV Extended, and is a resident program on the CDC 6600 computer system at Wright-Patterson AFB. Input to DLS consists of: 1) A description of the circuit elements and their interconnections; 2) Initial conditions and delay times; and 3) An input sequence of logic levels (Ref 2:1). For detailed operating instructions of the DLS program, the reader is referred to the DLS User's Manual.

The DLS program is used extensively here to verify design modifications before they are physically realized. In the case of the original MAPU design, the circuitry already exists and the outputs it generates are known. By describing the circuit in DLS language, the input sequence can be adjusted until the observed outputs match those expected of the circuit. With a valid set of inputs which simulate actual operation, the modified circuitry under test can be adjudged until the outputs once again match the desired results. This is the approach which is used throughout the testing. Of the three variables which make up the simulation (circuit, inputs, and outputs), only one will be unknown at any one time.

Original Circuit Model. Fig. 12 shows the output timing diagram resulting from the simulation of the circuit in Fig. 11. The program and tabular output can be found in Appendix A. This output accurately portrays the operation of the MAPU using ACP 127 message format (See Appendix B, Fig. 30). The inputs utilized are fixed as the input sequence for all further tests. The timing diagram shows the relationship between the inputs and the outputs. The program is used in the asynchronous mode of operation. In this mode the program reads a set of inputs, waits for all transients to die out, then prints outputs. Under this mode of operation all gate delay times are fixed at unity and the time axis has no significance except for reference to position. In all the simulations only the middle three input bits are shown on the output diagram. This is done to conserve space for more pertinent signals. Using the table of Baudot codes in Fig. 7, the reader can identify which input character is present by the middle three bits alone. The five bit Baudot code for a blank character is used as filler between flag characters. This filler represents the text of the message appearing between flags.

An examination of Fig. 12 reveals that it does in fact accurately represent the sequence of events expected from the analysis of the code recognition logic in Chapter III. At 0.65 usec, following the appearance of a CR and a LF, the SP character is detected on the trailing edge of the DREAD pulse. DETCO goes high and the $\overline{\text{BLIND}}$ signal goes low.

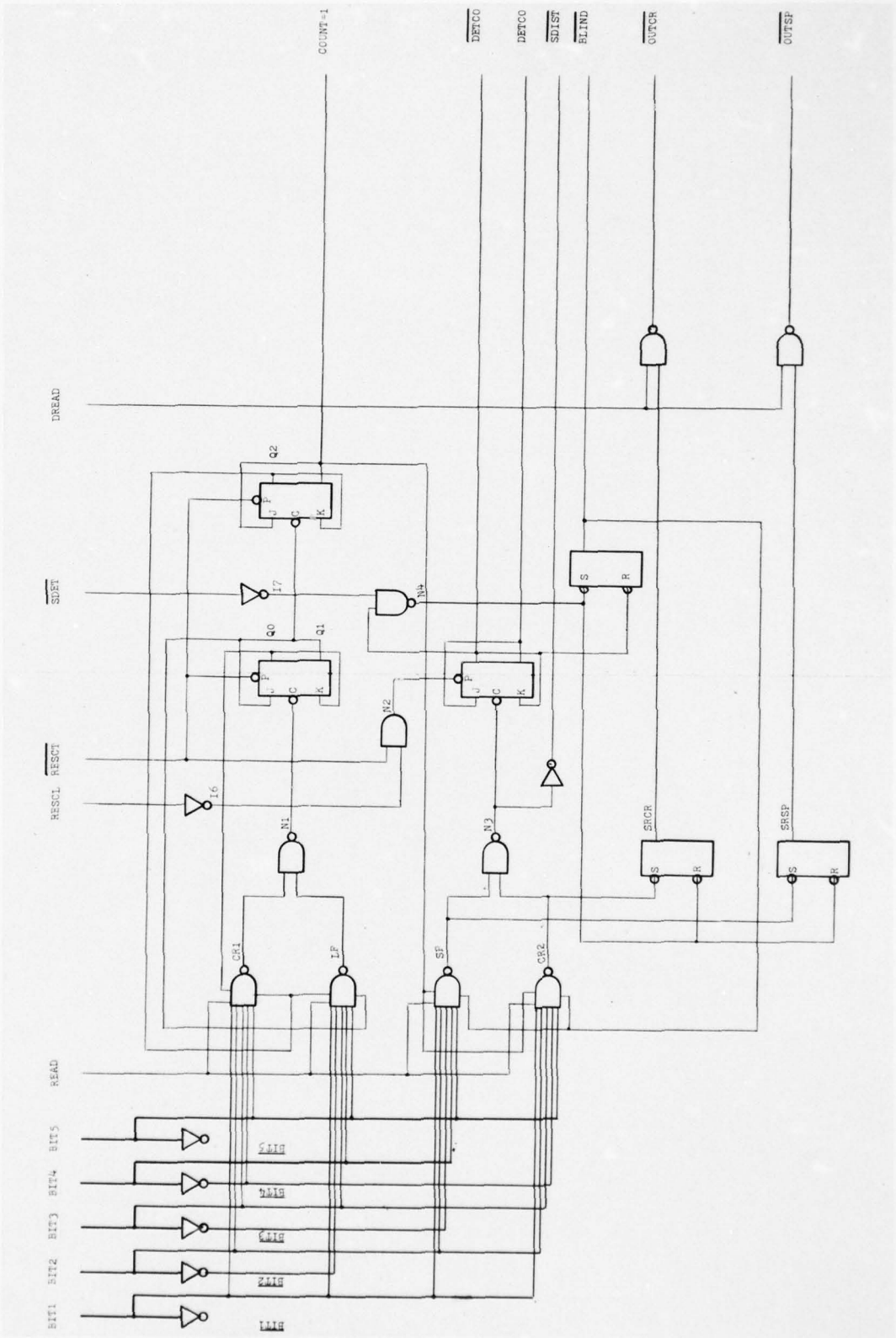


Fig. 11. Original Circuit DLS Model

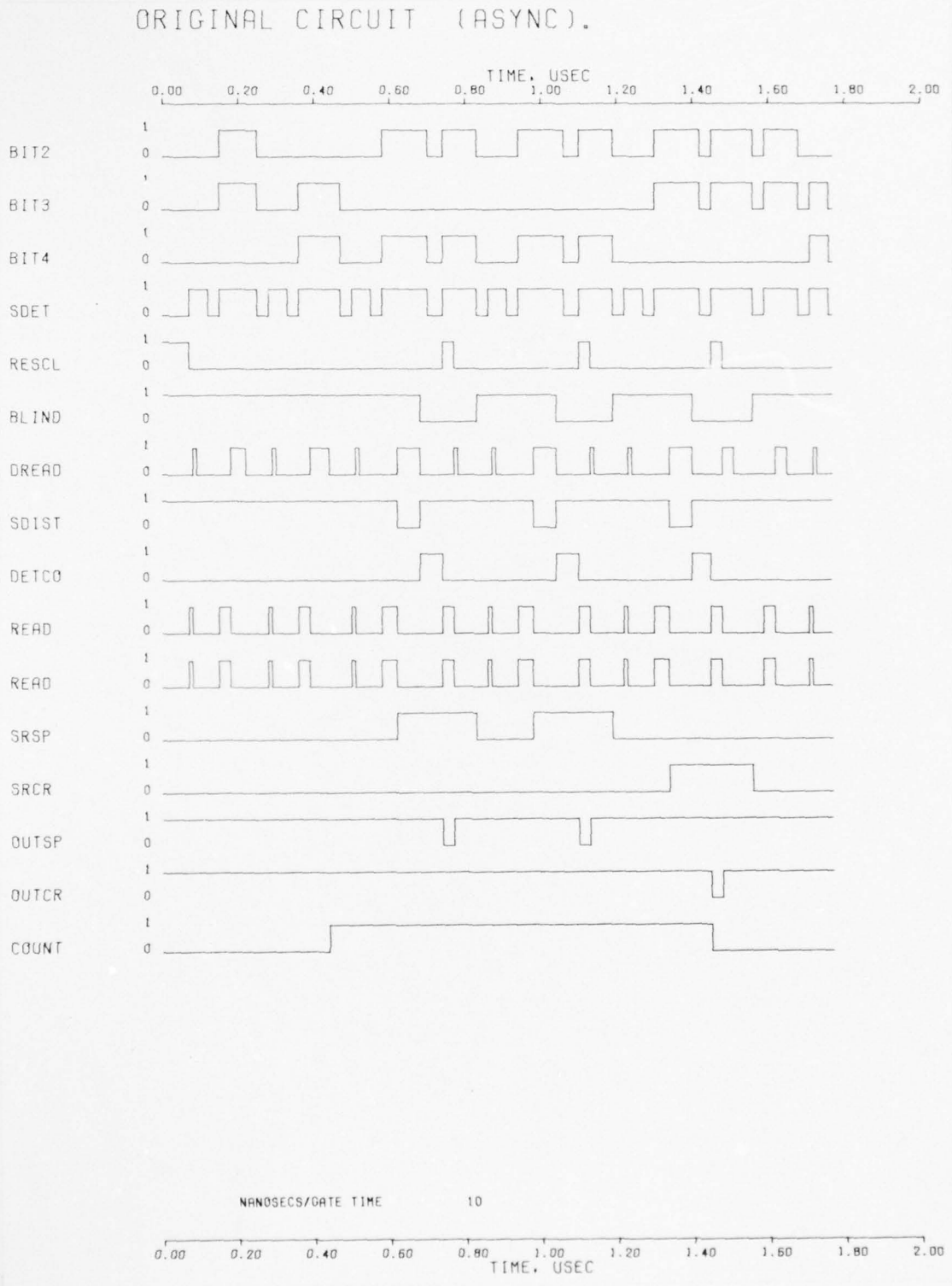


Fig. 12. Original Circuit Timing Diagram

The next READ pulse occurring at 0.75 usec causes a negative-going $\overline{\text{OUTSP}}$ pulse which reloads the SP character into the shift register. The input of a second SP at time 0.80 usec goes undetected because the $\overline{\text{BLIND}}$ signal is still low, inhibiting detection of the re-inserted character. The sequence is repeated for an additional SP and finally a CR character.

Modification 1

The first stage of modification of the code recognition logic is shown in Fig. 13. The circuit is similar to the original circuit of Fig. 11, with changes indicated by asterisks. The addition of the third JKMS flip-flop gives the circuit the ability to count down from an initialization level of four, instead of three as in the original logic. The choice of the JKMS device allows the counter to load its inputs when the clock pulse goes high and change output states when the pulse goes low. All the JKMS flip-flops are operated in the toggle configuration so that each clock pulse causes one change of state. The outputs of the flip-flops are used as enabling signals to a bank of AND gates, A1 through A8. These gates are enabled at appropriate times to pass the DREAD pulse from decoder gates FG, LT, SP, LF, and CR. The JKMS flip flop Q6 is added to store the occurrence of a CR character and enable the LF decoder gate. Set-Reset (SR) flip-flops SRLT and SRFG are added to store the occurrence of a LT and FG character respectively. Gates $\overline{\text{OUTLT}}$ and $\overline{\text{OUTFG}}$ are used in conjunction with the READ pulse to simulate the re-insertion of the stored character into the character

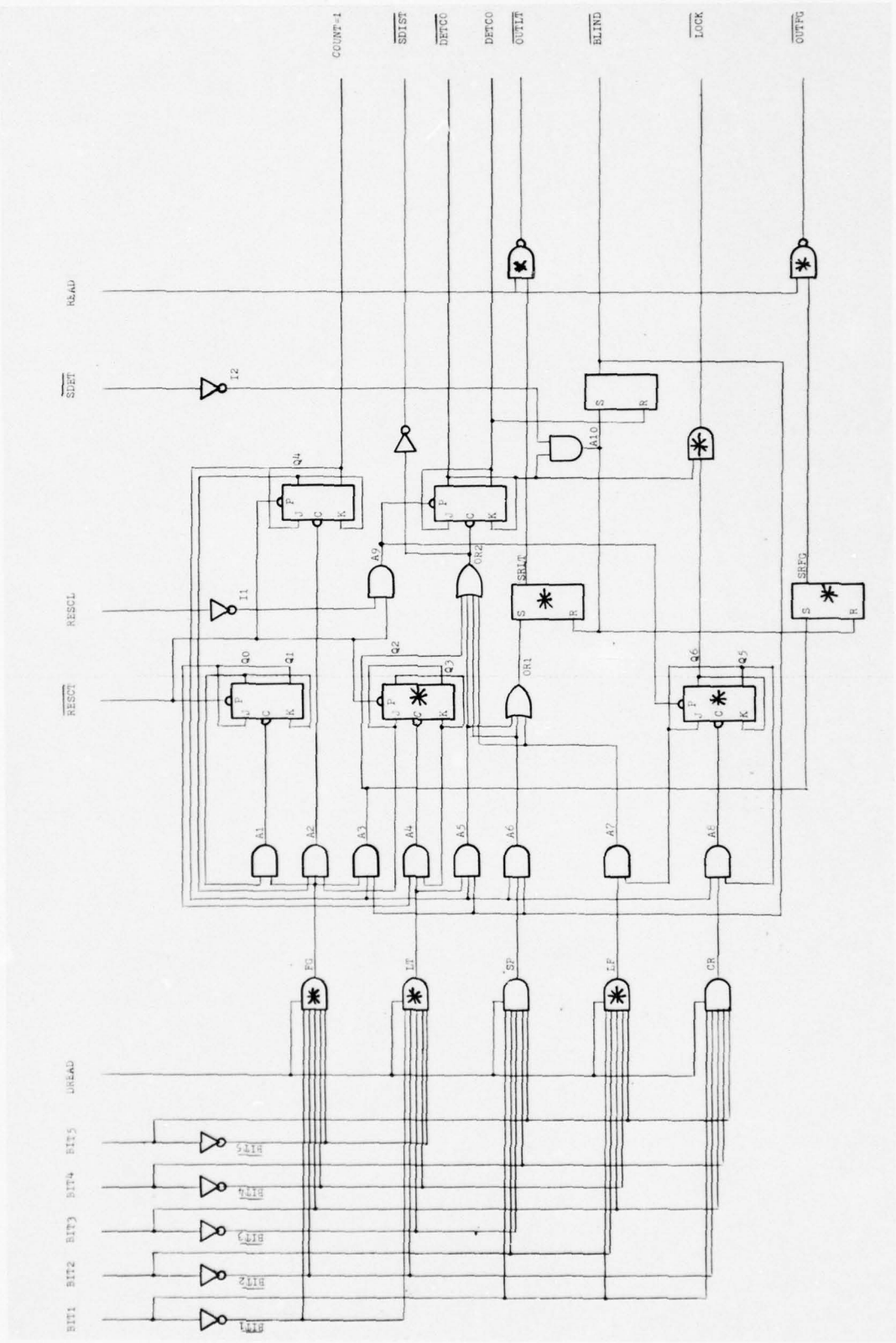


Fig. 13. Modification 1 DLS Model

distribution logic. AND gate $\overline{\text{LOCK}}$ is a new addition which generates the $\overline{\text{LOCK}}$ signal used to hold the output data buffer in the mark condition. This function used to be performed by the $\overline{\text{DETCO}}$ signal; however, as mentioned previously, the output buffer must now be disabled throughout the duration of the CRLF combination. AND gate $\overline{\text{LOCK}}$ is first pulled low by the change of state of the Q6 flip-flop, and is subsequently held low by the active $\overline{\text{DETCO}}$ signal upon detection of the LF. The data buffer is an integral part of the code recognition logic and is physically on the A2 Assembly; however, since it remains unchanged in this and all subsequent modifiactions it is not shown here. The reader is referred to Appendix C, Fig. 32.

Verification. The timing diagram resulting from the DLS simulation of the circuit appears in Fig. 14. The input sequence supplies the FG, LT, FG, LT combination of characters, interspaced with blank characters representing intervening portions of the tape header. The COUNT signal indicates that the logic is in the code count = 1 condition when the second FG is detected. The LT character is detected and the MAPU halts at approximately 0.90 usec on the time axis. The DETCO signal goes high and the $\overline{\text{LOCK}}$ signal goes low along with the $\overline{\text{BLIND}}$ signal. The next READ signal at 1.00 usec outputs a negative-going $\overline{\text{OUTLT}}$ pulse, representing the re-insertion of a LT character. The next set of characters is a CRLF combination. The first CR occurs at 1.30 usec. The $\overline{\text{LOCK}}$ signal goes low on the trailing edge of the DREAD signal which passes the character through the CR gate. The LF

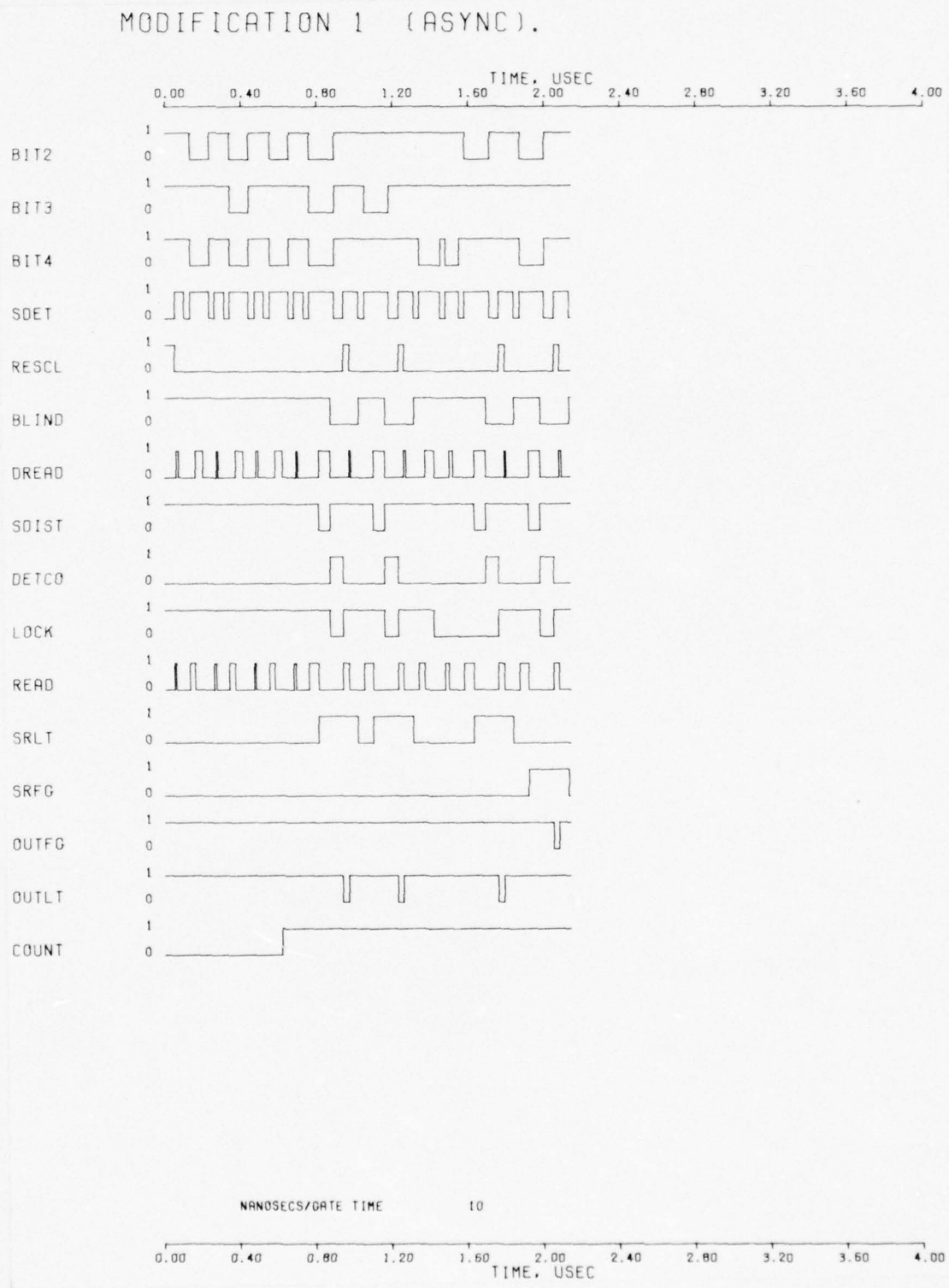


Fig. 14. Modification 1 Timing Diagram

decoder gate is now enabled to detect the LF at 1.60 usec. When the DREAD pulse goes low, the DETCO signal goes high and the normal halting sequence occurs. The last character in the input sequence is a FG. The trailing edge of the DREAD pulse at 2.00 usec sets DETCO high and the halt and storage cycle occurs again. Pulse $\overline{\text{OUTFG}}$ is generated coincident with the next READ pulse at 2.10 usec.

Modification 2

The second stage of the design process is shown in Fig. 15. Eight-input NAND gates are used to combine the five bit code and enabling signals. A four bit shift register, marked with asterisks, is used this time to follow the progression of the input character sequence. The outputs of the register are fed back to the decoder gates in straight and inverted form to perform the enabling function. The shift register is incorporated into this modification to compare its usefulness with that of the three-stage counter used in Modification 1. While this register performs the required function properly, no advantage in its use is found, and in fact it requires four additional inverters to develop the required signals inherently available at the $\overline{Q}$ outputs of the flip-flop counter. Also marked with asterisks are the additional gates A5 and $\overline{\text{OUTSP}}$, and the SR flip-flop SRSP. AND gate A5 is used in conjunction with the new input signal $\overline{\text{HALT}}$ to simulate the signal generated by the activation of the ITEM SELECT buttons. SR flip-flop SRLT is set upon detection of a LT, SP, or LF character. If the ITEM SELECT button is depressed for the

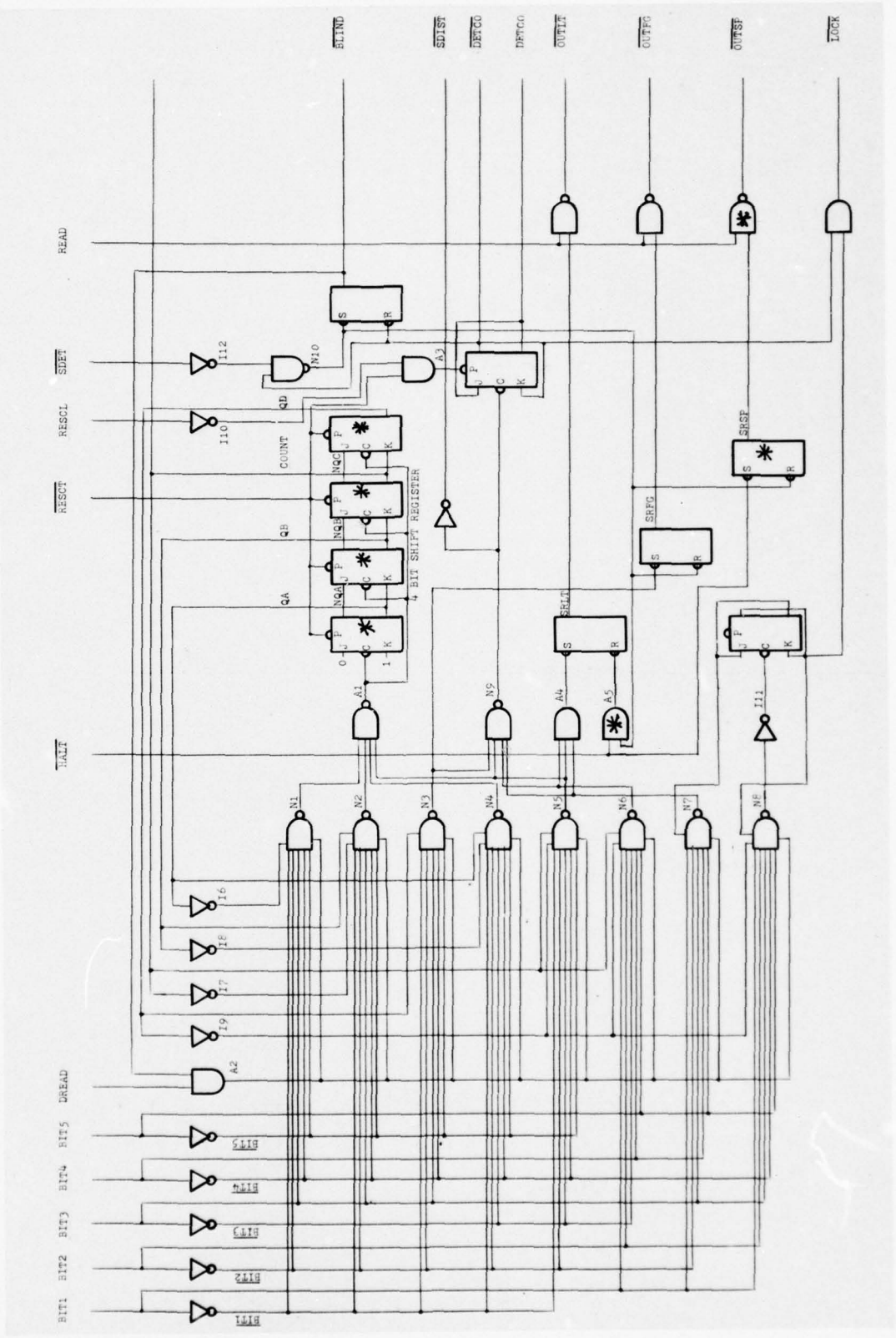


Fig. 15. Modification 2 DLS Model

second or subsequent time the $\overline{\text{HALT}}$ signal simultaneously resets SRLT and sets SRSP. SR flip-flop SRSP and NAND gate $\overline{\text{OUTSP}}$ are added to store the SP signal and output a negative-going pulse in conjunction with the READ signal. The $\overline{\text{HALT}}$ pulse is added to the input data sequence to approximate the appearance of a signal generated by ITEM SELECT button monitor circuitry.

Verification. The DLS timing diagram for the circuit of Fig. 15 appears in Fig. 16. The MAPU operates as in Modification 1 until time point 2.00 usec. The code recognition logic detects a SP character and sets the LT flip-flop. This is the desired sequence of events unless the ITEM SELECT button has been depressed once already. The appearance of the $\overline{\text{HALT}}$ signal at 2.00 usec indicates that this is in fact the case. The negative-going $\overline{\text{HALT}}$ pulse clears the SRLT flip-flop and sets the SRSP flip-flop just prior to the next READ pulse. The READ pulse occurring at 2.10 usec generates the $\overline{\text{OUTSP}}$ signal, simulating the re-insertion of the SP character into the shift register. The remainder of the operation proceeds as the previous design stage.

A flaw in the design of the first two stages is noted during this simulation. At time 2.50 usec the occurrence of a CR character is causing the $\overline{\text{LOCK}}$ signal to go low on the trailing edge of the DREAD pulse, but the CR is loaded into the shift register on the leading edge. For a very short period of time the character is in the shift register while the data buffer is unlocked. This sequence of events could lead to erratic behavior if one or more of the character bits

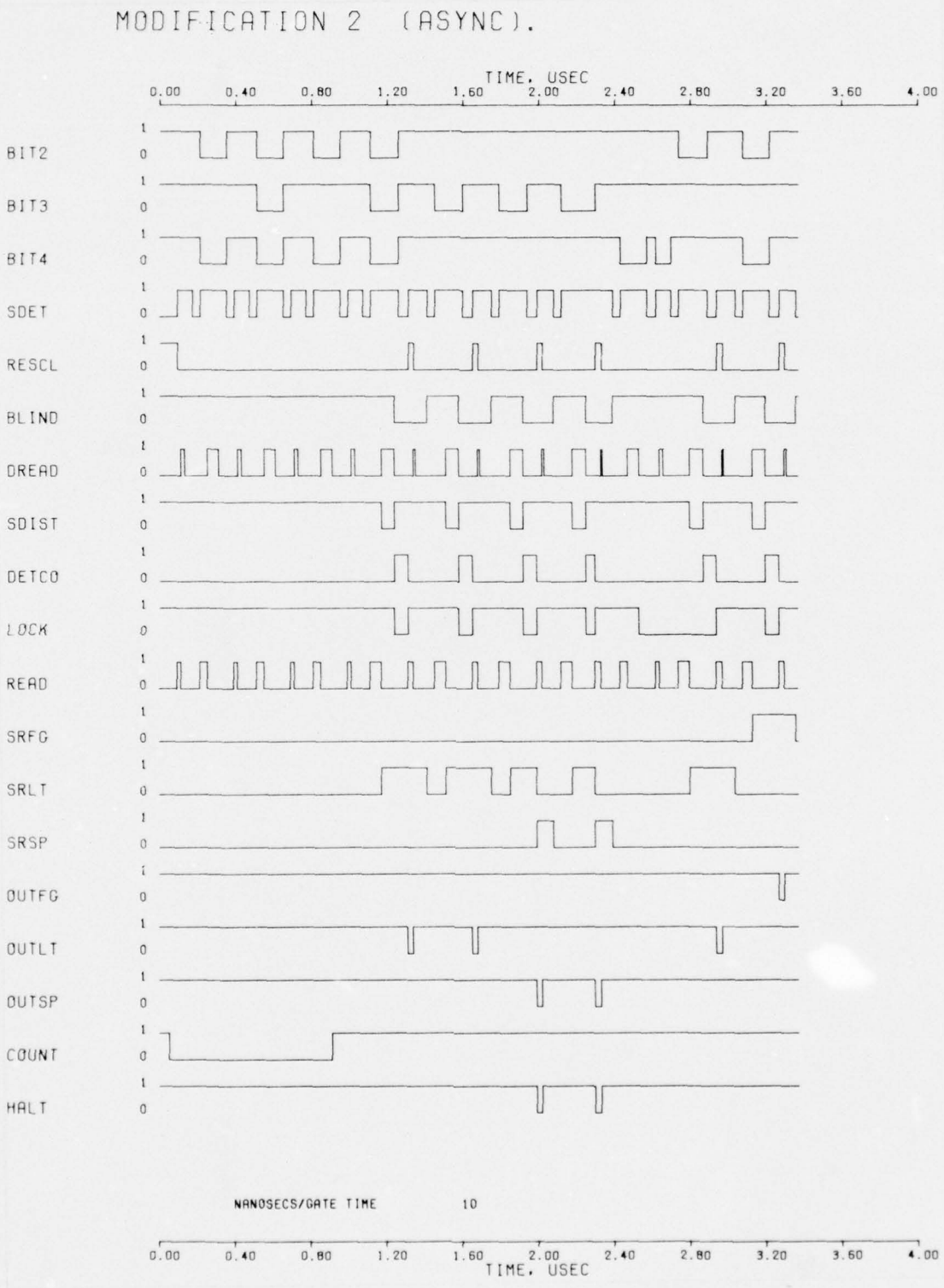


Fig. 16. Modification 2 Timing Diagram

in the register are shifted onto the data bus. The JKMS flip-flop implemented in the first stage of the design to detect the CR and enable the LF decoder gate is not changing state until the DREAD pulse is complete. This is the expected performance of the JKMS flip-flop and its choice is a design error (Ref 6:232-233). It must be replaced by a flip-flop which reacts to the leading edge of an input pulse. The SR flip-flop is such a device and it is incorporated into the next level of design.

Modification 3

Fig. 17 shows the final design stage. This design incorporates the changes made in the previous two stages and adds some additional circuitry to make the design as close to operational as possible. The START button is cross-connected in order to perform the functions of both the START and BROADCAST buttons. Gates N8, N9, and $\overline{\text{BROAD}}$ are added to allow for manual or automatic indexing into broadcast mode. When JKMS flip-flop Q3 changes state upon detection of the final LT character, NAND gate N8 can pass the START pulse if the operator elects to manually index to broadcast mode. The resulting negative pulse from N8 drives AND gate $\overline{\text{BROAD}}$ low and this signal sets the BROADCAST flip-flop on the A3 Assembly (Ref 1:6-3). If the code recognition logic is allowed to run full course through the routing indicator section of the message tape, the detection of the FG character preceeding the terminating period causes flip-flop Q2 to change states. The resulting high $\overline{Q}$ output enables

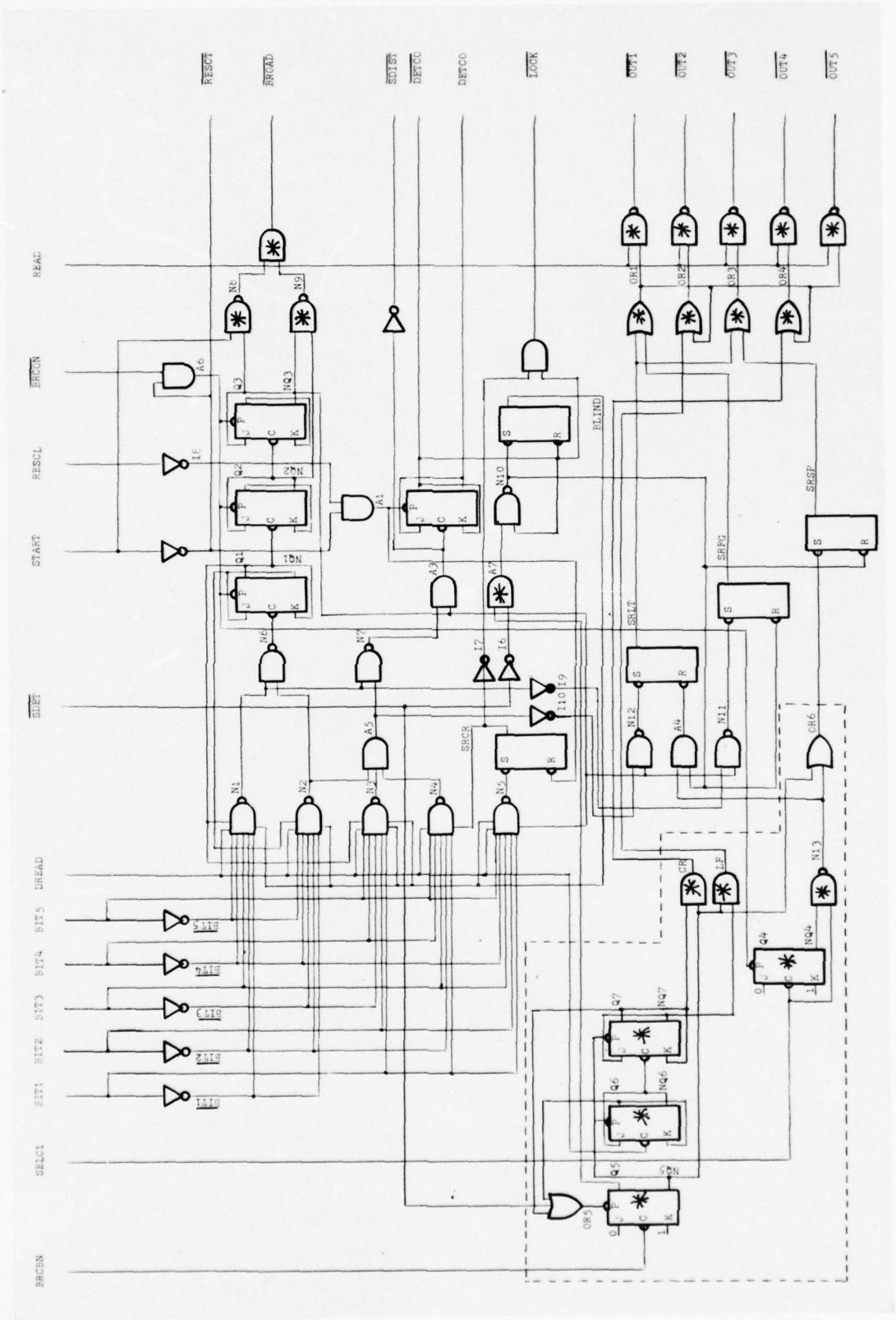


Fig. 17. Modification 3 DLS Model

NAND gate N9, and the result is automatic generation of the $\overline{\text{BROAD}}$ signal.

The number of decoder gates have been reduced to five, N1 through N5. These gates are alternately enabled by the Q and $\overline{Q}$ outputs of the Q1 flip-flop. The output encoder section, shown with asterisks, converts the contents of the storage flip-flops into complete five bit Baudot code characters for re-insertion.

The ITEM SELECT button monitor formed by flip-flop Q4 is the logic discussed in the previous model as generating the $\overline{\text{HALT}}$ signal. In this case the signal N13 forms the internal version of $\overline{\text{HALT}}$. This signal resets SRLT, and sets SRSP unless inhibited by the CRLF injector. The signal SELC1 is derived from the *ITEM SELECT button #1. The actual modification* would require six identical logic circuits, one for each button, to monitor the button usage. Signals SELC1 through SELC6 are the only signals which do not appear on the code recognition card Assembly A2.

The CRLF injector is formed by an unlocking flip-flop Q5, and two cascaded, toggling flip-flops Q6 and Q7. If the operator wishes to indicate an end of a line on one of the slave reperforators, he would activate the BROADCAST button, free now to be used to unlock the CRLF injector. Flip-flop Q5 changes state, freeing flip-flops Q6 and Q7 to toggle in response to successive DREAD pulses. When the CRLF injector is unlocked, NQ5 inhibits the setting of the SRSP flip-flop when the ITEM SELECT button is activated. Instead, AND gate

CR is activated and the first READ pulse following ITEM SELECT injects a CR character. DREAD immediately follows READ and toggles Q6 once. AND gate CR is still high and a second CR is injected. The following DREAD toggles Q6 once again, causing Q7 to toggle as well. NQ7 is now high and activates AND gate LF. The READ pulse injects the LF character and OR gate OR5, enabled by the low Q6 and Q7 signals, passes the next $\overline{\text{STOP DET}}$ pulse. This negative-going pulse presets flip-flop Q5, locking the CRLF injector. The low Q5 signal during the injection interval disables AND gate A7 and inhibits the setting of the BLIND flip-flop. Therefore, the decoding gates are disabled during the three-character interval, and the TD drive logic is blocked for the same period. When the CRLF injector is locked again, the Q5 signal goes high, the BLIND flip-flop is set, the decoder gates are enabled, and the TD driver logic resumes stepping the tape reader.

Verification. The timing diagram of the final design stage appears in Fig. 18. Sheet 1 shows the circuit utilized to perform routing segregation. Action up until 1.60 usec is the same as the previous stages. The output encoder now shows all five bits of the re-inserted character each time the READ pulse occurs. At 1.60 usec the operator activates the BROADCAST button (BRCBN) before activating the #1 ITEM SELECT button (SELC1). The next three characters injected up through time period 2.20 usec are two CRs and one LF. The $\overline{\text{BLIND}}$ signal is low throughout this three-character interval to

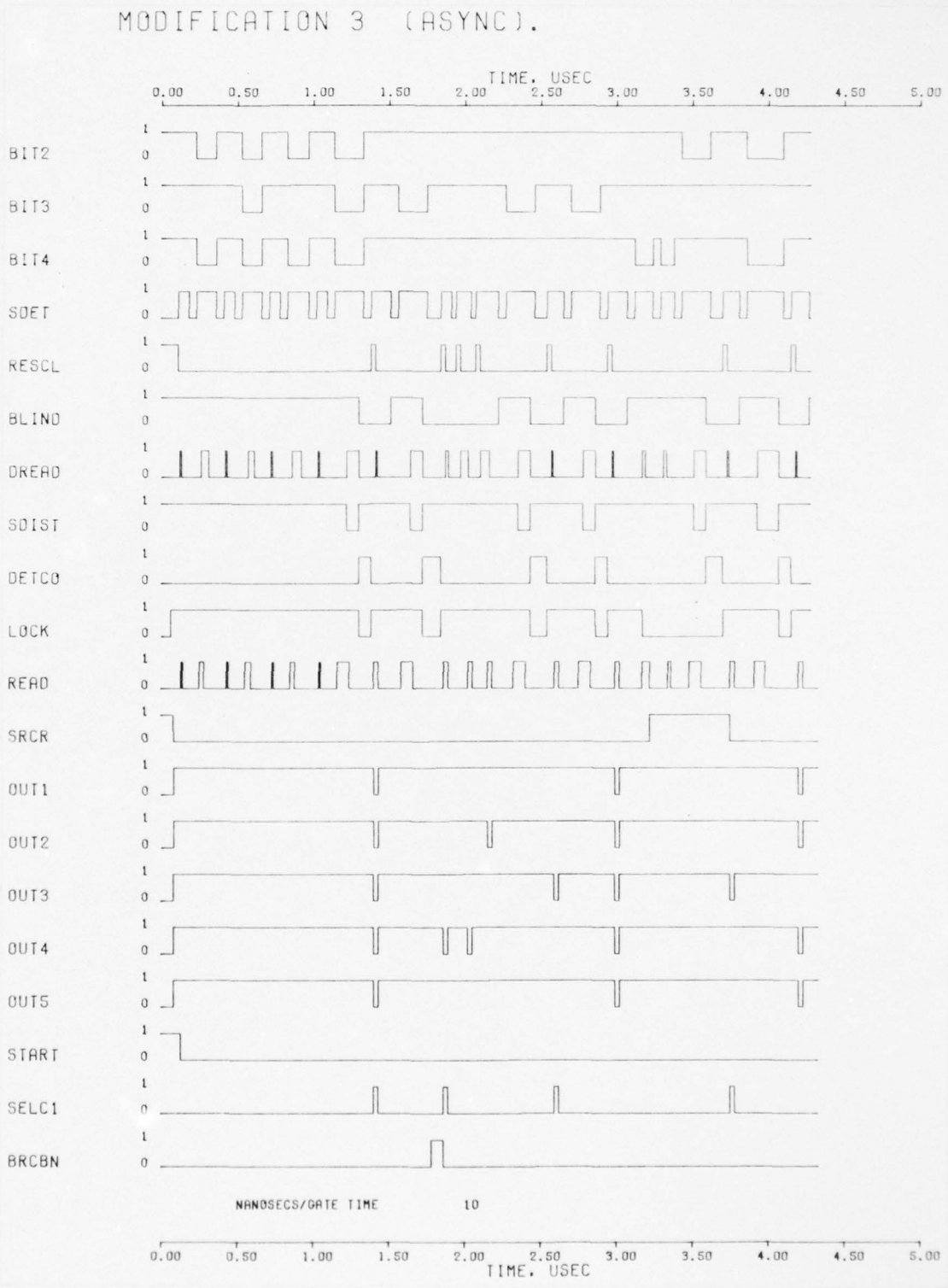


Fig. 18. Modification 3 Timing Diagram (Sheet 1 of 2)

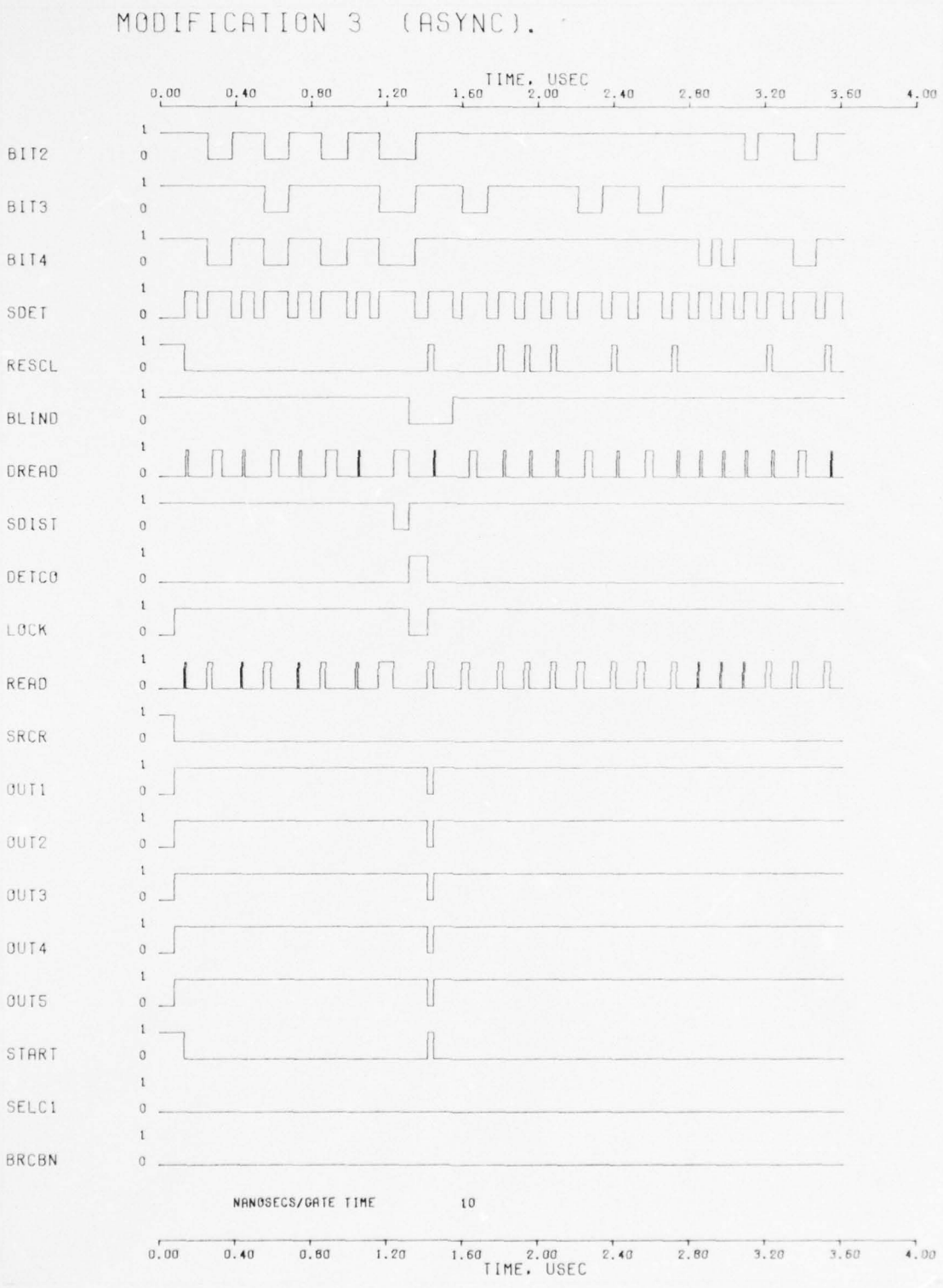


Fig. 18. Modification 3 Timing Diagram (Sheet 2 of 2)

inhibit stepping of the tape reader. The detection of the FG character at time 4.40 usec automatically indexes the code count, and the output encoder injects a FG character at the next READ pulse.

Sheet 2 of Fig. 18 shows the Modification 3 circuit used as a tape duplicator. At time 1.50 usec the START button is depressed a second time, and code recognition logic is indexed into broadcast mode, and the stored LT character is re-inserted on the next READ pulse.

The addition of the SR flip-flop in place of the JKMS flip-flop as a CR detector corrects the flaw noted in Modification 2. At time period 3.20 usec the $\overline{\text{LOCK}}$ signal goes low on the leading edge of the DREAD signal as specified.

Modification 3 is the final stage of design before hardware realization. Chapter V discusses the construction and testing of the hardware based on the results of this chapter.

V. HARDWARE CONSTRUCTION, TESTING, AND RESULTS

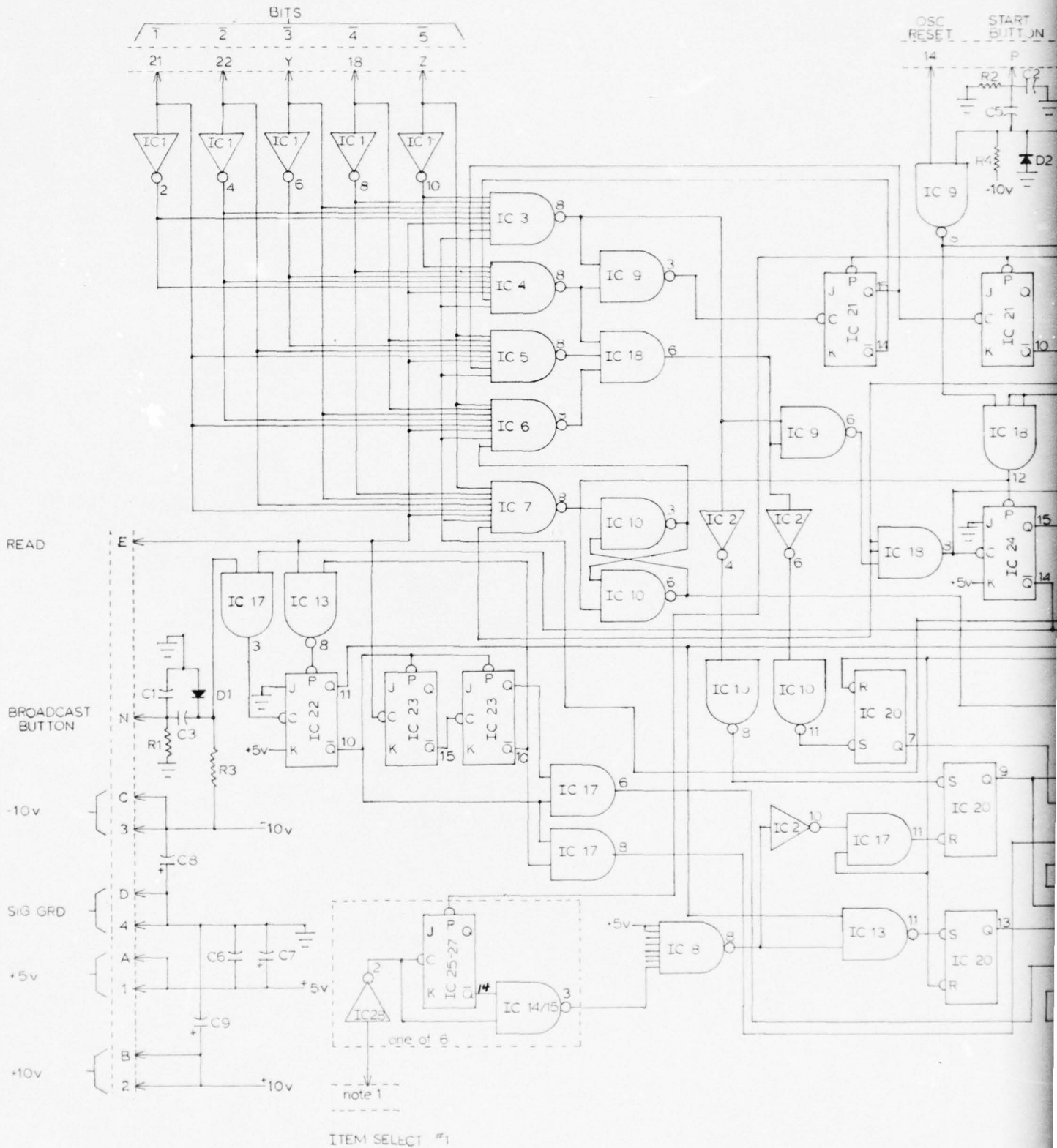
Hardware Construction

Modification 3 in Chapter IV incorporates all the desired design features specified by the sponsoring agency. A hardware realization of the computer model shown in Fig. 18 must now be built. Construction constraints set by AFCS include low cost and readily available parts which meet MIL-STD-883. The SN 5400 series integrated circuitry made by Texas Instruments meets both of the requirements listed above. In addition these integrated circuits (ICs) are completely documented in Reference 5, and a good cross-section supply is maintained by the AFIT supply system. The SN 5400 series assures reliable operation over a thermal range of $-55^{\circ}C$ to $125^{\circ}C$. Since the 5400 series employs Transistor-to-Transistor Logic (TTL), it is fully compatible with the Diode-to-Transistor Logic (DTL) used in the construction of the original MAPU circuitry. The logic voltage range for DTL is 0 to +5 volts relative to the ground plane. The range for TTL is 0 to +3.5 volts, but both logic families share a switching point of 1.5 volts $\pm$0.1 volt. The power supply in the 1A2 Assembly provides a highly regulated source of 5 volts DC (Ref 1:6-7). The DTL and TTL families have a typical fan-out figure of from 8 to 10, and power dissipation per gate is roughly the same at 10 milliwatt for standard TTL. Typical gate delay times for DTL are 30 nsec, as opposed to 10 nsec for TTL (Ref 6:86-88). This added speed is advantageous because the modified code recognition

logic is more sophisticated than the original circuitry. With an increased number of levels of gates, signal propagation times increase. With the added speed of TTL, the increased complexity is not a problem (Ref 6:83).

The A2 circuit card which the modification replaces is approximately 12 inches long and 4 inches wide. The card is double-sided and terminates in 44 gold-plated pins, numbered 1 through 22 and lettered A through Z. Because it is impractical to build and test a circuit on a finished printed circuit card, a design breadboard is used. The modification prototype utilizes an Elite 3 design board made by the firm of EL Instruments, Inc. of Derby, Connecticut. This breadboard has five banks of molex IC sockets and a row of ten indicator lamps and toggle switches. Power is provided by the circuit under test, and in this case is supplied by the pin connections of the 1A1 Assembly (Ref 1:6-8). An interconnect harness constructed with blank 22-pin circuit cards and 14-conductor flat lead is used to connect the design board to the card receptacle in the 1A1 Assembly.

The hardware realization shown in Fig. 19 is the design which resulted from the modification stages of Chapter IV. In order to pass the additional signals from the code recognition logic board (A2) to the character distribution logic board (A1), three additional jumper connections are made at the back of the 1A1 Assembly. These jumper connections are shown in Fig. 20. The isolation diodes allow a single NAND gate to pull Bit 1 and Bit 5 low without allowing these bit signals to influence each other. An auxiliary cable harness



ITEM SELECT #1

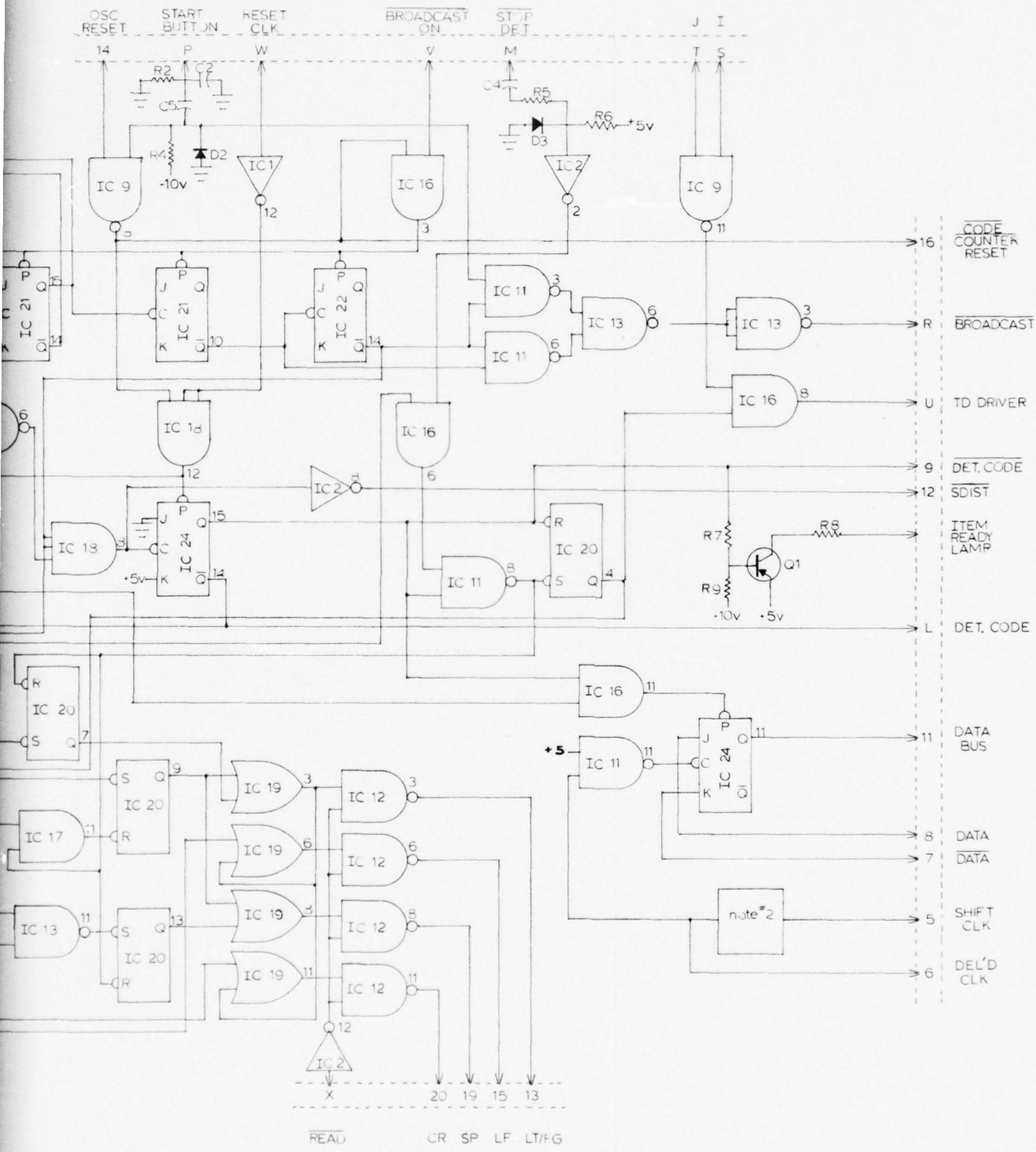


Fig. 19. Modified Code Recognition Logic (Sheet 1 of 2)

Notes:

1. An auxiliary cable harness with six conductors is used to route the six ITEM SELECT button pulses from the A3 Assembly.

2. The delay circuitry of the original A2 Assembly. See Appendix C, Fig. 32.

3. Unless otherwise indicated, resistances are in ohms, 1/4 watt, ± 5%. Capacitances are in pf.

4. All diodes are 1N4148.

5. Q1 is a 2N2905.

6. Integrated Circuits:

   a. IC1, IC2, and IC28 are SN5404. Pin 7 is Gnd. Pin 14 is + 5 volts.

   b. IC3, IC4, IC5, IC6, IC7, and IC8 are SN5430. Pin 7 is Gnd. Pin 14 is + 5 volts.

   c. IC9, IC10, IC11, IC12, IC13, IC14, and IC15 are SN5400. Pin 7 is Gnd. Pin 14 is + 5 volts.

   d. IC16 and IC17 are SN5408. Pin 7 is Gnd. Pin 14 is + 5 volts.

   e. IC18 is SN5411. Pin 7 is Gnd. Pin 14 is + 5 volts.

   f. IC19 is SN5432. Pin 7 is Gnd. Pin 14 is + 5 volts.

   g. IC20 is SN54279. Pin 8 is Gnd. Pin 16 is + 5 volts.

   h. IC21, IC22, IC23, IC24, IC25, IC26, and IC27 are SN5476. Pin 13 is Gnd. Pin 5 is + 5 volts.

Fig. 19. Modified Code Recognition Logic (Sheet 2 of 2)

is required to pass the six ITEM SELECT button signals from the A3 Assembly to the A2 Assembly. The connections on the A3 card are made to IC9-3, IC9-6, IC9-8, IC9-11, IC10-8, and IC10-11. The connection points are marked with asterisks on the A3 circuit diagram, Appendix C, Fig. 33. These signals are connected to the six identical ITEM SELECT monitors in the modified logic.

The ICs of the circuit are interconnected with #22 solid hook-up wire. Care is required to lay out the wire runs, not only to provide a neat appearance, but to keep stray capacitance and noise generation down to a minimum. Several of the most pertinent signals are routed to indicator lamps on the

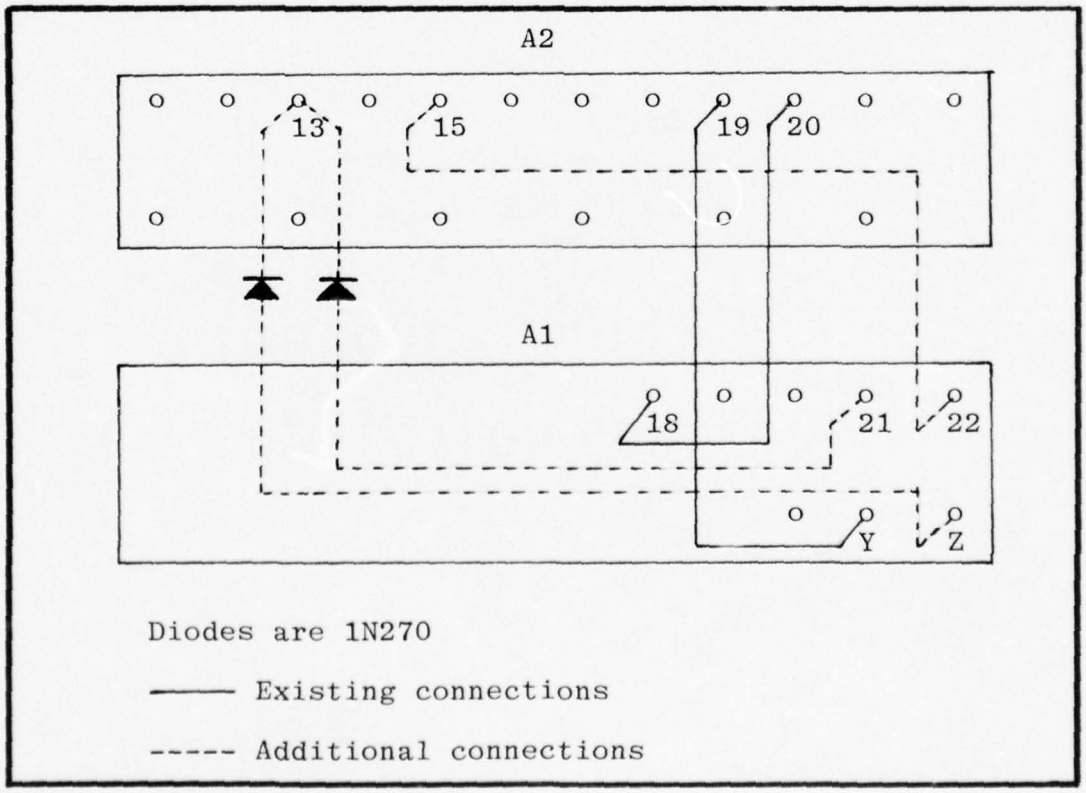


Fig. 20. 1A1 Assembly Jumper Connections

design board in order to monitor the behavior of the design while under test. The signals monitored include $\overline{\text{BLIND}}$, DETCODE, $\overline{\text{LOCK}}$, and the status of the code counter and storage flip-flops. Parts of the original code recognition circuitry are used unchanged in the modified design. These portions are the 0.75 bit interval delay circuit, and the differentiators for the START, BROADCAST, and $\overline{\text{STOP DET}}$ signals. Because these circuits perform properly as originally designed, they were adopted in the modification. Appendix C, Fig. 32 shows these portions in detail and is referred to by the notes in Fig. 19.

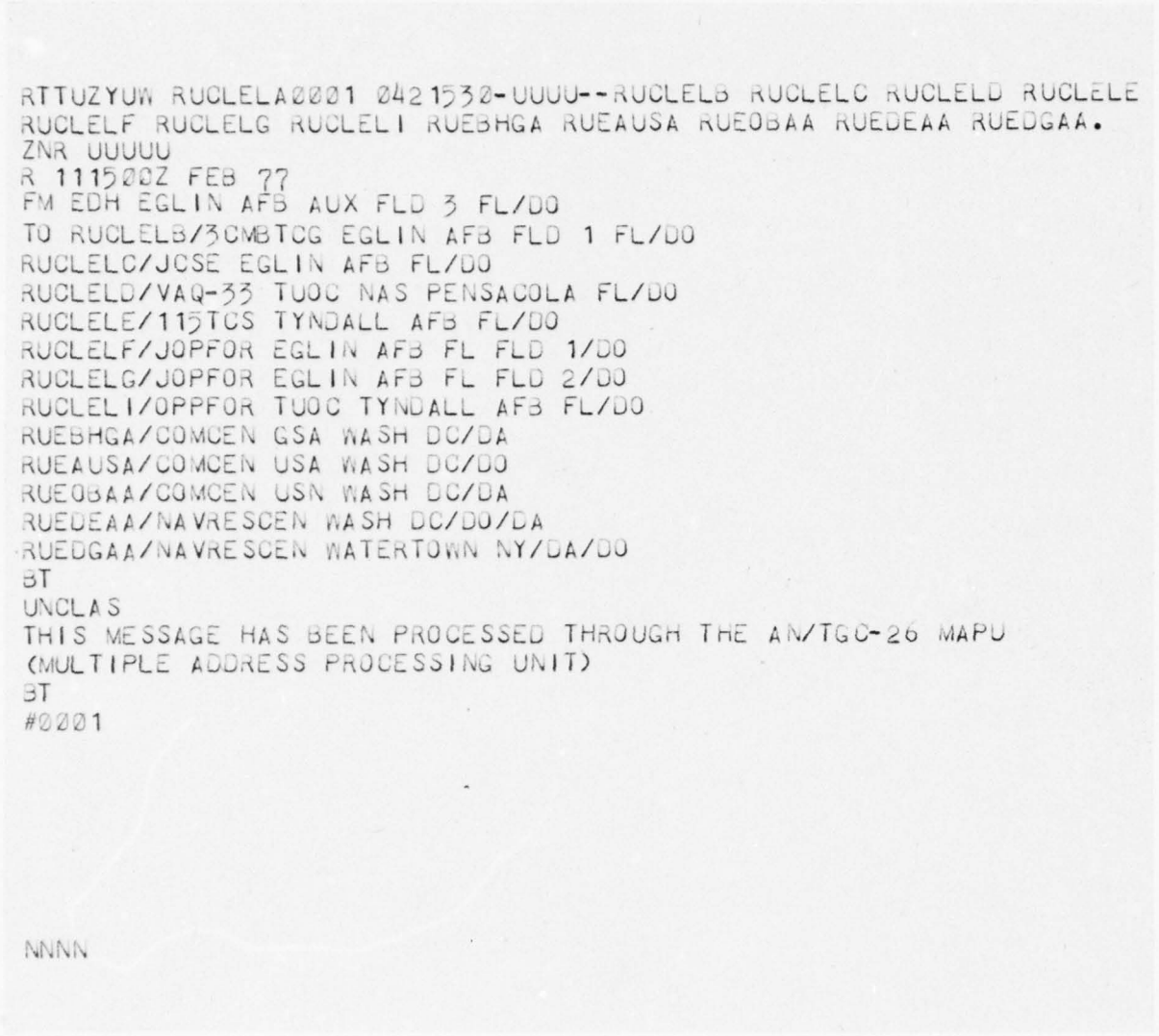
Testing

The MAPU modification testing was carried out at the Wellesley Air National Guard (ANG) Station, Massachusetts, using the equipment and personnel of the 253rd CMBTCG. On November 11th, 1976 a preliminary version of the design was tested. This version closely approximated the Modification 1 computer model shown in Fig. 13. The primary purpose of this test was to check the validity of the computer modeling, and to verify that the newly designed LOCK function would operate with sufficient speed to freeze the data bus before the contents of the shift register were transmitted. Using a test tape prepared by TSGT Cummings, an active duty enlisted advisor to the ANG unit, the design board was interfaced with the MAPU within the AN/TGC-26. The action of the design was interpolated from the indications of the display lamps on the design board and the position of the tape reader at each halt.

It was a tribute to the accuracy of the DLS simulations that the design performed properly on the first trial. There was no tape output during this test because the design at this stage lacked output encoder and CRLF injector circuitry. During this visit the decision to use the BROADCAST button in a new role as the CRLF injector button was reached after discussion of the problem with ANG communications personnel.

On February 10th, 1977, the design underwent final testing at Wellesley. The circuit of Fig. 19 was completely realized on the design board. Fig. 21 shows a page copy of the test tape that was used to evaluate the MAPU performance. The tape contains a standard message heading, including originating station routing indicator and station serial number. The routing indicator section contains 12 routers, and there is a short text in the body of the message. Using this tape, the MAPU was activated and the routing indicators were segregated in a number of different ways calculated to demonstrate all the capabilities of the new code recognition logic. The MAPU was also utilized as a tape duplicator by depressing the START button a second time when the MAPU tape reader reached the first halt.

The MAPU was operated for a period of approximately two hours on the first day. One of the operators suggested that a possibility of operator error existed in the CRLF injector. If the operator inadvertently depressed the BROADCAST button at any time during MAPU operation, an unwanted CRLF combination would be transmitted to the slave tape. NAND gate IC17-3 was installed in place of an inverter to lock out the



```
RTTUZYUW RUCLELA0001 0421530-UUUU--RUCLELB RUCLELC RUCLELD RUCLELE
RUCLELF RUCLELG RUCLELI RUEBHGA RUEAUSA RUEOBAA RUEDEAA RUEDGAA.
ZNR UUUUU
R 111500Z FEB 77
FM EDH EGLIN AFB AUX FLD 3 FL/DO
TO RUCLELB/3CMBTCG EGLIN AFB FLD 1 FL/DO
RUCLELC/JCSE EGLIN AFB FL/DO
RUCLELD/VAQ-33 TUOC NAS PENSACOLA FL/DO
RUCLELE/115TCS TYNDALL AFB FL/DO
RUCLELF/JOPFOR EGLIN AFB FL FLD 1/DO
RUCLELG/JOPFOR EGLIN AFB FL FLD 2/DO
RUCLELI/OPPFOR TUOC TYNDALL AFB FL/DO
RUEBHGA/COMCEN GSA WASH DC/DA
RUEAUSA/COMCEN USA WASH DC/DO
RUEOBAA/COMCEN USN WASH DC/DA
RUEDEAA/NAVRESCEN WASH DC/DO/DA
RUEDGAA/NAVRESCEN WATERTOWN NY/DA/DO
BT
UNCLAS
THIS MESSAGE HAS BEEN PROCESSED THROUGH THE AN/TGC-26 MAPU
(MULTIPLE ADDRESS PROCESSING UNIT)
BT
#0001



NNNN
```

Fig. 21. JANAP 128 Master Test Tape Page Copy

BROADCAST button except during those time periods when the MAPU was stopped following the detection of a code character. At that time signal DETCODE from IC24-14 would enable the NAND gate. This time period is the only legal injection point of the CRLF combination. With this change in the circuitry, the design was activated again on February 11th and operated for approximately three hours. During this time a representative of AFCS/DOOT was present to observe the MAPU operation.

Reliability

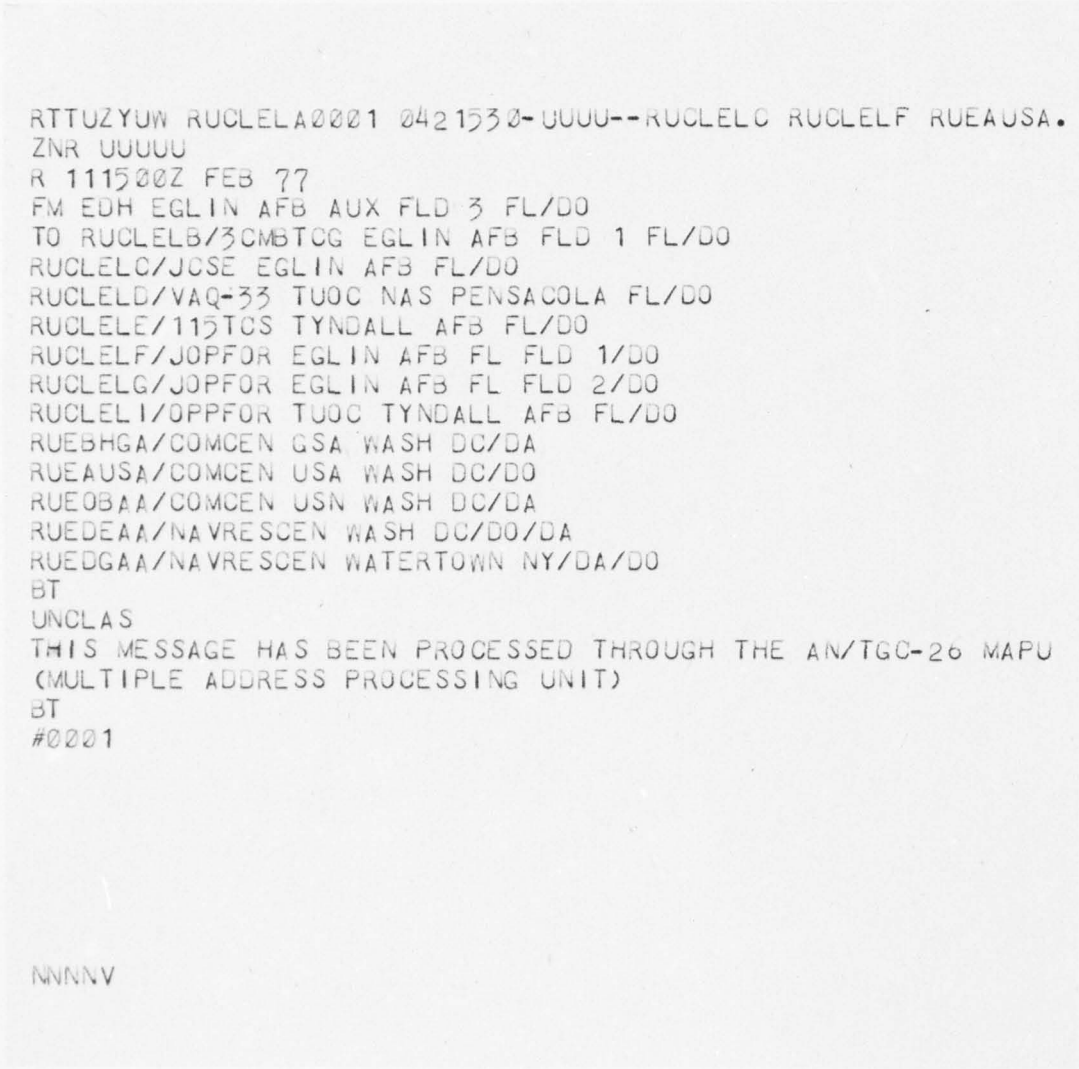
The prototype modification should present no unusual reliability problems. This circuit tested at Wellesley had been burned-in for approximately 20 hours prior to test. During the more than five hours of live testing the circuit displayed no erratic behavior. Power supply voltage measurements revealed no degradation in the voltage regulation, despite the increased power consumption of the prototype. The advantages of TTL logic have been discussed previously. One disadvantage of this logic family is an increased sensitivity to input noise (Ref 6:87). The live testing at Wellesley turned up no problems due to noise and it should be pointed out that this was a worst-case situation. The interconnection was over three feet long and made of small gauge wire. The hook-up wire on the breadboard itself was loose and of varying lengths. The possibilities of attenuation, cross-talk, and noise were certainly much higher in the prototype circuit than on a properly etched circuit card. The maximum fan-out

figure for the circuit is nine for IC16-3. This figure is within the limits for TTL logic (Ref 6:87). A potential dynamic hazard exists in the operation of IC24-15 in the toggle mode. If this flip-flop oscillates momentarily in changing states, it can cause a partial blanking of the data bus. In order to minimize this possibility the flip-flop is operated with the J input grounded and the K input tied to a source of +5 volts to insure a stable change of states when the flip-flop is clocked. No other area is seriously threatened by a potential hazard unless the logic were to slow to the level of DTL, a factor of two or three.

Results

The results of the second day of testing appear in Fig. 22. This figure shows page copy of three slave tapes generated by the modified MAPU using the master tape of Fig. 21. Sheet 1 and 2 show a slave message of three routing indicators each, randomly segregated from the master tape. Sheet 3 contains a slave message of six randomly segregated routing indicators and demonstrates the injection of the CRLF combination after four routers on the first line. Each of the slave tapes has a LT character re-inserted properly after the double "dash" combination, and all tapes terminate the routing indicator section correctly with a period.

The results of the testing carried out in February fulfilled all the requirements set by AFCS to the satisfaction of all representatives of the 253rd CMBTCG and AFCS present at the demonstration. The only recommendation for improvement

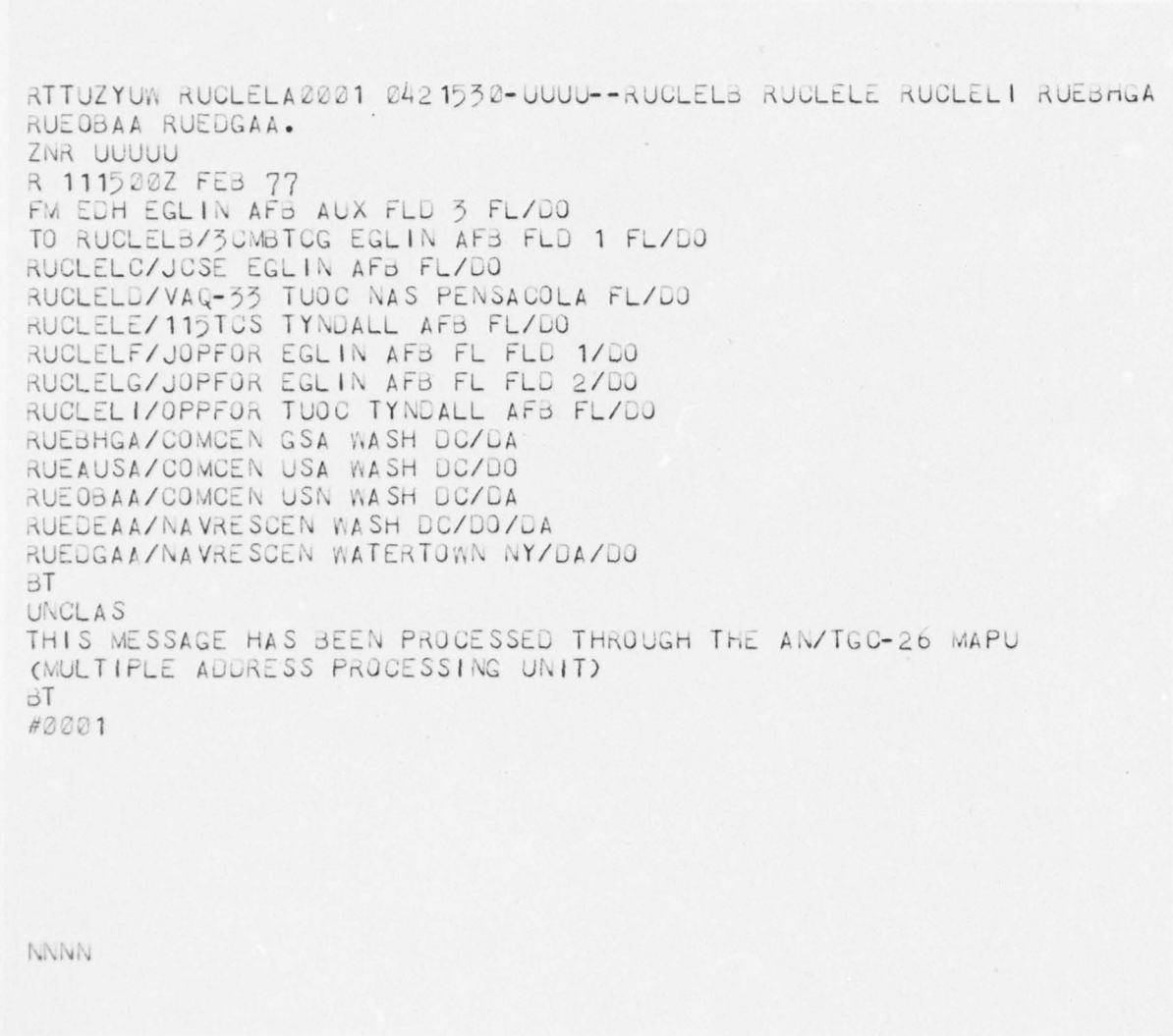


```
RTTUZYUW RUCLELA0001 0421530-UUUU--RUCLELC RUCLELF RUEAUSA.
ZNR UUUUU
R 111500Z FEB 77
FM EDH EGLIN AFB AUX FLD 3 FL/DO
TO RUCLELB/3CMBTCG EGLIN AFB FLD 1 FL/DO
RUCLELC/JCSE EGLIN AFB FL/DO
RUCLELD/VAQ-33 TUOC NAS PENSACOLA FL/DO
RUCLELE/115TCS TYNDALL AFB FL/DO
RUCLELF/JOPFOR EGLIN AFB FL FLD 1/DO
RUCLELG/JOPFOR EGLIN AFB FL FLD 2/DO
RUCLELI/OPPFOR TUOC TYNDALL AFB FL/DO
RUEBHGA/COMCEN GSA WASH DC/DA
RUEAUSA/COMCEN USA WASH DC/DO
RUEOBAA/COMCEN USN WASH DC/DA
RUEDEAA/NAVRESCEN WASH DC/DO/DA
RUEDGAA/NAVRESCEN WATERTOWN NY/DA/DO
BT
UNCLAS
THIS MESSAGE HAS BEEN PROCESSED THROUGH THE AN/TGC-26 MAPU
(MULTIPLE ADDRESS PROCESSING UNIT)
BT
#0001



NNNNV
```

Fig. 22. JANAP 128 Slave Tape Page Copy (Sheet 1 of 3)

```
RTTUZYUW RUCLELA0001 0421530-UUUU--RUCLELD RUCLELG RUEDEAA.
ZNR UUUUU
R 111500Z FEB 77
FM EDH EGLIN AFB AUX FLD 3 FL/DO
TO RUCLELB/3CMBTCG EGLIN AFB FLD 1 FL/DO
RUCLELC/JCSE EGLIN AFB FL/DO
RUCLELD/VAQ-33 TUOC NAS PENSACOLA FL/DO
RUCLELE/11STCS TYNDALL AFB FL/DO
RUCLELF/JOPFOR EGLIN AFB FL FLD 1/DO
RUCLELG/JOPFOR EGLIN AFB FL FLD 2/DO
RUCLELI/OPPFOR TUOC TYNDALL AFB FL/DO
RUEBHGA/COMCEN GSA WASH DC/DA
RUEAUSA/COMCEN USA WASH DC/DO
RUEOBAA/COMCEN USN WASH DC/DA
RUEDEAA/NAVRESCEN WASH DC/DO/DA
RUEDGAA/NAVRESCEN WATERTOWN NY/DA/DO
BT
UNCLAS
THIS MESSAGE HAS BEEN PROCESSED THROUGH THE AN/TGC-20 MAPU
(MULTIPLE ADDRESS PROCESSING UNIT)
BT
#0001

NNNNV
```

Fig. 22. JANAP 128 Slave Tape Page Copy (Sheet 2 of 3)



```
RTTUZYUW RUCLELA0001 0421530-UUUU--RUCLELB RUCLELE RUCLELI RUEBHGA
RUEOBAA RUEDGAA.
ZNR UUUUU
R 111500Z FEB 77
FM EDH EGLIN AFB AUX FLD 3 FL/DO
TO RUCLELB/3CMBTCG EGLIN AFB FLD 1 FL/DO
RUCLELC/JCSE EGLIN AFB FL/DO
RUCLELD/VAQ-33 TUOC NAS PENSACOLA FL/DO
RUCLELE/115TCS TYNDALL AFB FL/DO
RUCLELF/JOPFOR EGLIN AFB FL FLD 1/DO
RUCLELG/JOPFOR EGLIN AFB FL FLD 2/DO
RUCLELI/OPPFOR TUOC TYNDALL AFB FL/DO
RUEBHGA/COMCEN GSA WASH DC/DA
RUEAUSA/COMCEN USA WASH DC/DO
RUEOBAA/COMCEN USN WASH DC/DA
RUEDEAA/NAVRESCEN WASH DC/DO/DA
RUEDGAA/NAVRESCEN WATERTOWN NY/DA/DO
BT
UNCLAS
THIS MESSAGE HAS BEEN PROCESSED THROUGH THE AN/TGC-26 MAPU
(MULTIPLE ADDRESS PROCESSING UNIT)
BT
#0001



NNNN
```

Fig. 22. JANAP 128 Slave Tape Page Copy (Sheet 3 of 3)

received involved the replacement of the CRLF injector button (BROADCAST button) with a more sophisticated version of the ITEM SELECT button monitor which would automatically inject the CRLF combination after the first four routers and every succeeding eight. This suggestion is investigated in Chapter VI.

VI. RECOMMENDATIONS

Design Improvements

One of the goals outlined in Chapter I is the elimination of all unnecessary manual interface with the MAPU during operation. The pursuit of this goal is tempered by the cost and size contraints set by the sponsoring agency. The circuit shown in Fig. 19 of Chapter V is an example of this clash between design improvements and physical constraints. The use of a manual push button to inject the CRLF combination is an obvious second choice to an automatic injection system which would monitor the use of all six ITEM SELECT buttons and insert the CRLF at the correct point on each slave tape. Such an ITEM SELECT monitor is of necessity more complex than the simple monitor circuit used. Its complexity is such that there is not sufficient space on the A2 Assembly to house the new code recognition logic and the more sophisticated monitor logic. The goal of this chapter is the investigation of improvements which might be incorporated in the design if one or more of the sponsor's constraints are removed or relaxed.

Automatic CRLF Injection

As the code recognition logic is currently modified, the operator must monitor the number of routing indicators on each slave tape. When the number of routers reaches four the operator depresses the CRLF push button before transmitting a fifth routing indicator to a given slave reperforator.

This is a task which can readily be performed by a logic circuit since the line spacing in JANAP 128 is such that there is room for up to four routing indicators on the first line, and eight on every line thereafter. The fact that these numbers do not change from situation to situation makes it easy to design a circuit to sense the end of the line. The DLS circuit of Fig. 23 is such a device. The circuit is made up of a three-stage binary counter and combinational logic to sense the count = 4 condition. Since the counter counts up to decimal seven before starting over at zero, the combinational logic is triggered at the end of the first four clock pulses, and every eight pulses thereafter. The logic of Fig. 23 contains the CRLF injector developed in the Modification 3 design stage, and the entire diagram replaces that portion of Fig. 17, Chapter IV, outlined by a dashed line. Flip-flop Q4 performs as before to sense the first use of the ITEM SELECT button, and to set the SP flip-flop on all subsequent uses. JKMS flip-flops Q8, Q9, and Q10 form the three-stage counter, clocked by the ITEM SELECT signal. AND gate A8 is connected to Q outputs Q8, Q9, and NQ10, as well as gate A3. When the counter reaches state 1 1 0, A8 is enabled and can pass the positive pulse from gate A3. A3 produces a pulse when a flag character is detected, and the resulting signal from A8 unlocks the CRLF injector just as the manual push button would. The timing table in Fig. 24 shows the automatic CRLF injector in use with the circuit from Modification 3. At approximately 3.20 usec a fifth SELC1 pulse appears, simulating the fifth time that an ITEM SELECT button is depressed.

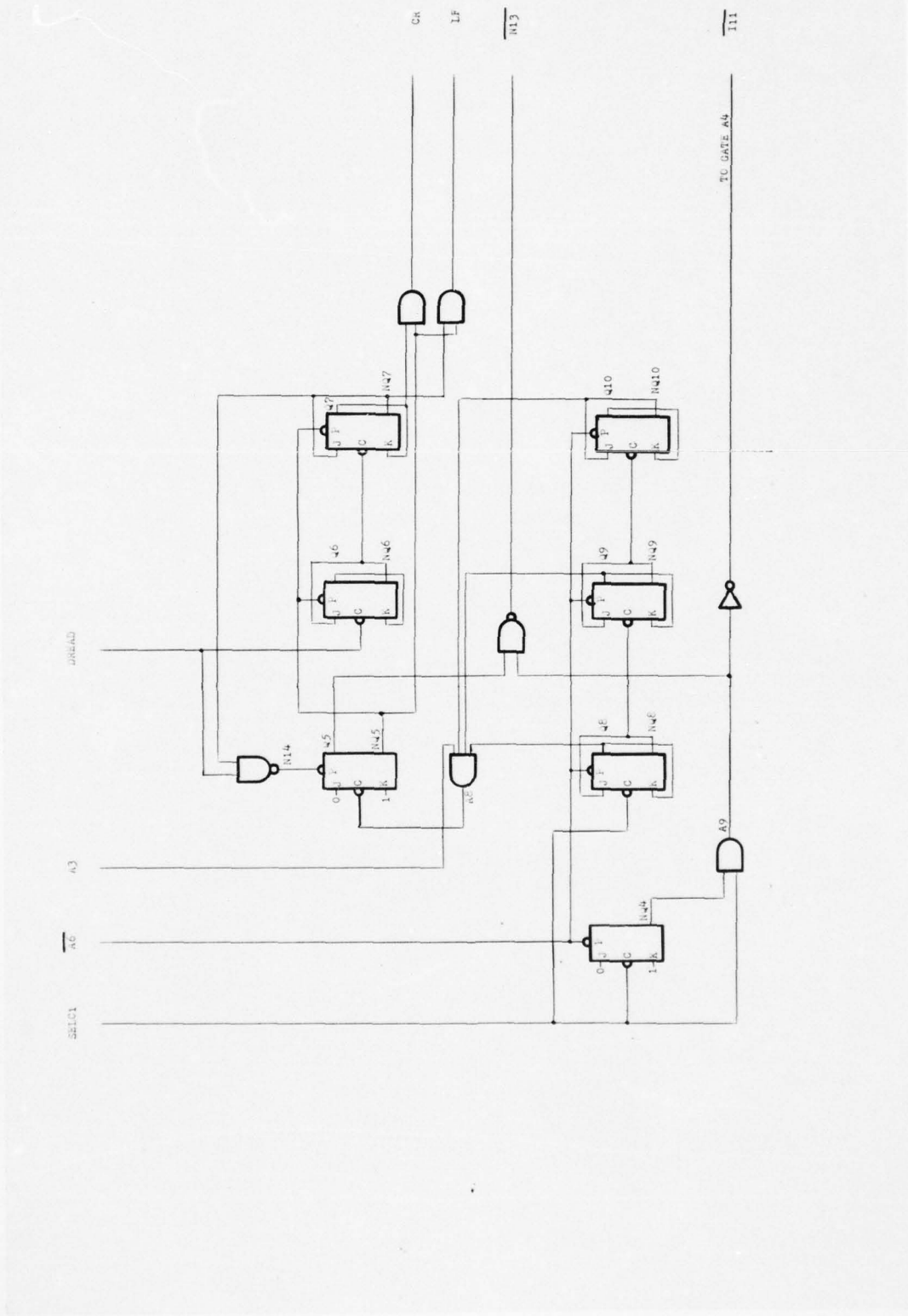


Fig. 23. Automatic CRLF Injector

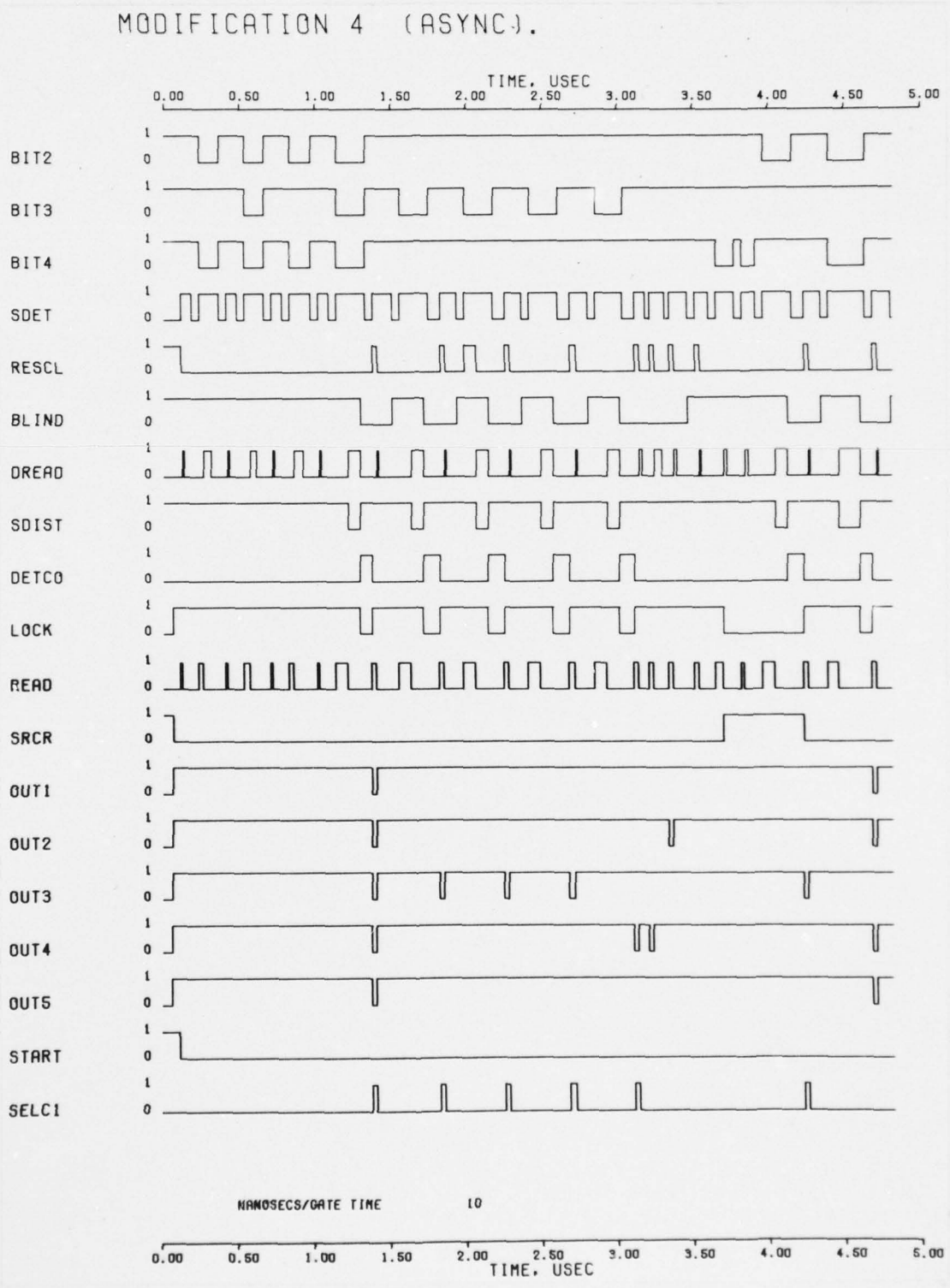


Fig. 24. Modification 4 Timing Diagram

Corresponding to the SELC1 pulse are two CRs and one LF occuring at 3.20, 3.30, and 3.40 usec respectively. The signal BRCBN, representing the BROADCAST button, has been eliminated entirely.

As can be seen, the automatic CRLF injector requires substantially more gates than the manual injection version. Using off the shelf integrated circuits and keeping in mind that six identical circuits must be built, one for each button, the total number of integrated circuit packages on the A2 card increases by approximately 12. This number exceeds the physical space limitations of the circuit card. If at some future time the cost constraint is relaxed, the function of the six ITEM SELECT monitors can easily be built into one specially ordered LSI package.

Circuit Production

Chapter I listed the AFCS requirement that the MAPU retain the capability of processing ACP 127. There are two approaches to producing the code recognition logic modification which meet this constraint. The modification can be built on a single circuit card, physically compatible with the 1A1 Assembly in which it is mounted. This card could be supplied to AN/TGC-26 using units where field technicians could then perform the additional wiring shown in Fig. 20. The six-conductor harness interconnecting the A2 and A3 Assemblies would also be installed in the field. This harness requires quick-disconnect plugs on each end so that the original A2 circuit card can be installed for ACP 127 operation.

This approach requires that each unit using the AN/TGC-26 stock two different A2 card Assemblies. The advantage of this approach is low production costs, while a disadvantage is the requirement of further modification by using units.

The other approach is to produce entire 1A1 drawer Assemblies at the depot level with the modified A2 card installed and all additional interconnections made. This approach increases initial costs, but it has the advantage of quick changeover in the field and does not require attention on the part of the using unit. Each AN/TGC-26 would carry two 1A1 Assemblies, one for each format.

Circuit Simplification

As already mentioned, space is at a premium on the A2 circuit card. One way to increase available space and simplify the circuit is to use Read-Only Memories (ROMs) in the decoder and encoder sections of the code recognition logic. A Programmable Read-Only Memory (PROM) such as the Texas Instruments SN54186, used as an input decoder, could replace the five bit inverters of IC1, along with gates IC3, IC4, IC5, IC6, IC7, IC9-3, IC9-6, IC2-4, IC2-6, IC10-3, IC10-6, IC18-6, and IC18-8 (Ref 5:405). By using the six binary select and two enable inputs, the PROM can be programmed to respond to the proper combination of five code bits, $\overline{\text{BLIND}}$, READ, and Q of IC21-15. Each flag character would have the effect of selecting an eight bit word out of the 64 binary words available. These eight bit words can be organized so as to supply the correct signals directly to the code counter, DET CODE

flip-flop, storage flip-flops, etc. Similarly a smaller ROM or PROM can be programmed to perform the function of the output encoder formed by IC12, IC19, and IC2-12. The use of just two ROMS or PROMs would effectively replace approximately $10\frac{1}{2}$ integrated circuit packages containing 27 discrete gates. Such replacement would afford a saving of 37% in total gates used. Once again the limiting factor in such improvments is the cost constraint. These suggested simplifications have been included in this study in the event that further slow-downs in the production of equipment to replace the AN/TGC-26 require upgrading of the MAPU (Ref 7:1).

BIBLIOGRAPHY

1. T.O. 31W4-2TGC26-12. Control Reperforator Set C-7818/TGC-26. Washington D.C.: United States Air Force, January 17, 1973 (Technical Order).

2. Niederhauser, J.R. Digital Logic Simulator. Masters Thesis, Air Force Insitute of Technology. December, 1971.

3. Cummings, J.F. TSGT, Air Force Advisor. Personal Conversation. November 11, 1976.

4. Zemple, C.J. Director of C-E Mobile Communications. AN/TGC-26 Message Handling Deficiencies. An Air Force Tactical Air Command letter dated May 7, 1974.

5. Texas Instruments Inc. The TTL Data Book. USA: Texas Instruments Inc. 1973.

6. Hill, F.J. and G.R. Peterson. Introduction to Switching Theory and Logical Design (Second Edition). New York: John Wiley and Sons, 1968

7. Morris, E.C. Asst DCS/Mobile Communications. Message Handling Deficiencies. An Air Force Tactical Air Command letter dated August 6, 1974.

APPENDIX A

Digital Logic Simulator Programs and Timing Table Outputs

The timing diagrams which appear in Chapter IV and VI were generated by the DLS programs in Figs 25 through 29 on the following pages. The timing table output contains the same information as the timing diagram but in tabular form. For more detailed information about the execution of the DLS program see Reference 2.

```
**********************
* PROGRAM DLS        *
* IS NOW BEING EXECUTED *
**********************

*DLS      ORIGINAL CIRCUIT  (ASYNC).                              00010
$PATCH
      INPUT BIT1  BIT2  BIT3  BIT4  BIT5  SDET  RESCL RESET DREAD READ
      PRINT BIT2  BIT3  BIT4  SDET  RESCL BLIND DREAD SDIST DETCO
      PRINT BIT1  SRSP  SRCP  OUTSP OUTCR COUNT RESET
IBIT1 INV   BIT1
IBIT2 INV   BIT2
IBIT3 INV   BIT3
IBIT4 INV   BIT4
IBIT5 INV   BIT5
I6    INV   RESCL
SDIST INV   N2
I7    INV   SDET
N1    NAND  CP1   LF
N2    AND   I6    RESET
N3    NAND  SP    CP2
CP1   NAND  BIT1  BIT2  BIT3  IBIT4 BIT5  DREAD 00    02
LF    NAND  BIT1  IBIT2 BIT3  BIT4  BIT5  DREAD 01    02
SP    NAND  BIT1  BIT2  IBIT3 BIT4  BIT5  DREAD COUNT BLIND
CR2   NAND  BIT1  BIT2  BIT3  IBIT4 BIT5  DREAD COUNT BLIND
N4    NAND  I7    NDETC
OUTSP NAND  SRSP  READ
OUTCR NAND  SRCP  READ
00    JKFF  01    00    N1                01          RESET
02    JKFF  COUNT 02    01                COUNT       RESET
NDETC JKFF  DETCO NDETC N3                DETCO       N2
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SRSP  NAND  SP    A   A
A   A NAND  N4    SRSP
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SRCP  NAND  CP2   A   B
A   B NAND  N4    SRCP
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
BLIND NAND  N4    A   C
A   C NAND  NDETC BLIND
*****PRECEDING LINES PRODUCED BY GENDEK*****
$MODE ASYNC
$INITIAL CONDITIONS
CP1   1
CP2   1
SP    1
LF    1
N1    1
OUTSP 1
OUTCR 1
BLIND 1
00    1
02    1
NDETC 1
N1    1
SDIST 1
$RUN        10
```

Fig. 25. Original Circuit DLS Program (Sheet 1 of 3)

BEST AVAILABLE COPY

TIMING TABLE-- ORIGINAL CIRCUIT (ASYNC).

TIME	BIT3	BIT2	BIT1	BIT0	RESET	BLIND	DREAD	SDEST	DETCD	READ	SRSP	SRDR	OUTSP	OUTDR	COUNT	RESET
0	0	0	0	0	1	1	0	1	0	0	0	0	1	1	0	0
4	0	0	0	0	1	1	0	1	0	0	0	0	1	1	0	0
7	0	0	0	1	0	1	0	1	0	1	0	0	1	1	0	1
9	0	0	0	1	0	1	1	1	0	0	0	0	1	1	0	1
9	0	0	0	1	0	1	0	1	0	0	0	0	1	1	0	1
12	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	1
15	1	1	0	1	0	1	0	1	0	1	0	0	1	1	0	1
18	1	1	0	1	0	1	1	1	0	0	0	0	1	1	0	1
22	1	1	0	1	0	1	0	1	0	0	0	0	1	1	0	1
25	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	1
29	0	0	0	1	0	1	0	1	0	1	0	0	1	1	0	1
29	0	0	0	1	0	1	1	1	0	0	0	0	1	1	0	1
30	0	0	0	1	0	1	0	1	0	0	0	0	1	1	0	1
33	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	1
36	0	1	1	1	0	1	0	1	0	1	0	0	1	1	0	1
39	0	1	1	1	0	1	1	1	0	0	0	0	1	1	0	1
44	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1
47	0	0	0	0	0	1	0	1	0	0	0	0	1	1	1	1
50	0	0	0	1	0	1	0	1	0	1	0	0	1	1	1	1
51	0	0	0	1	0	1	1	1	0	0	0	0	1	1	1	1
52	0	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1
55	0	0	0	0	0	1	0	1	0	0	0	0	1	1	1	1
58	1	0	1	1	0	1	0	1	0	1	0	0	1	1	1	1
62	1	0	1	1	0	1	1	0	0	0	1	0	1	1	1	1
63	1	0	1	1	0	0	0	1	1	0	1	0	1	1	1	1
70	0	0	0	0	0	0	0	1	1	0	1	0	1	1	1	1
74	1	0	1	1	1	0	0	1	0	1	1	0	0	1	1	1
77	1	0	1	1	0	0	1	1	0	0	1	0	1	1	1	1
78	1	0	1	1	0	0	0	1	0	0	1	0	1	1	1	1
83	0	0	0	0	0	1	0	1	0	0	0	0	1	1	1	1
86	0	0	0	0	1	0	1	0	1	0	0	0	1	1	1	1
87	0	0	0	1	0	1	1	1	0	0	0	0	1	1	1	1
88	0	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1
91	0	0	0	0	0	1	0	1	0	0	0	0	1	1	1	1
94	1	0	1	1	0	1	0	1	0	1	0	0	1	1	1	1
98	1	0	1	1	0	1	1	0	0	0	1	0	1	1	1	1
104	1	0	1	1	0	0	0	1	1	0	1	0	1	1	1	1
105	0	0	0	0	0	0	0	1	1	0	1	0	1	1	1	1
110	1	0	1	1	1	0	0	1	0	1	1	0	0	1	1	1
113	1	0	1	1	0	0	1	1	0	0	1	0	1	1	1	1
114	1	0	1	1	0	0	0	1	0	0	1	0	1	1	1	1
119	0	0	0	0	0	1	0	1	0	0	0	0	1	1	1	1
122	0	0	0	1	0	1	0	1	0	1	0	0	1	1	1	1
123	0	0	0	1	0	1	1	1	0	0	0	0	1	1	1	1
124	0	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1
127	0	0	0	0	0	1	0	1	0	0	0	0	1	1	1	1
130	1	1	0	1	0	1	0	1	0	1	0	0	1	1	1	1
134	1	1	0	1	0	1	1	0	0	0	0	1	1	1	1	1
140	1	1	0	1	0	0	0	1	1	0	0	1	1	1	1	1
142	0	0	0	0	0	0	0	1	1	0	0	1	1	1	1	1
146	1	1	0	1	1	0	0	1	0	1	0	1	1	0	0	0

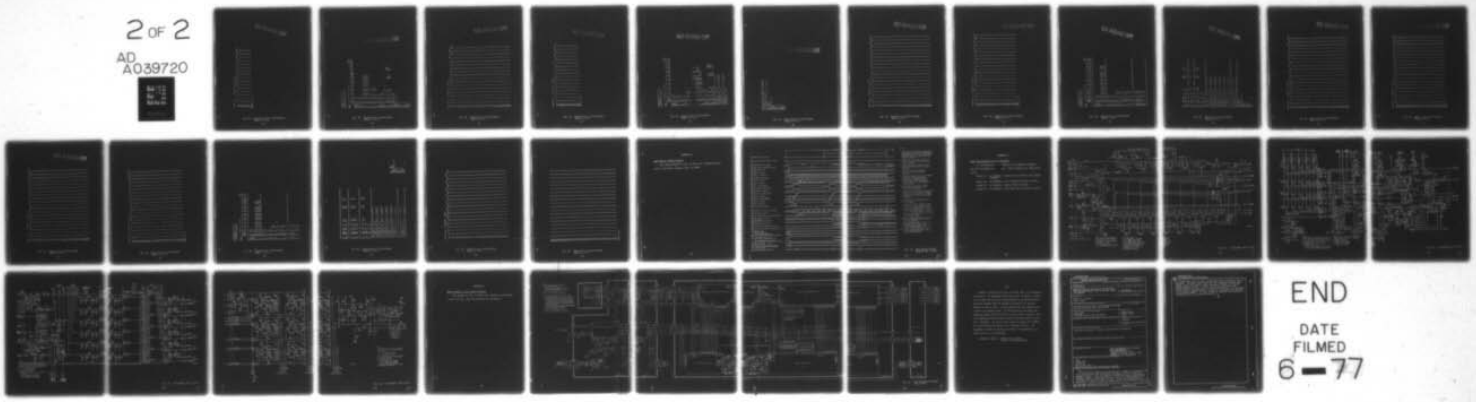
Fig. 25. Original Circuit DLS Program (Sheet 2 of 3)

BEST AVAILABLE COPY

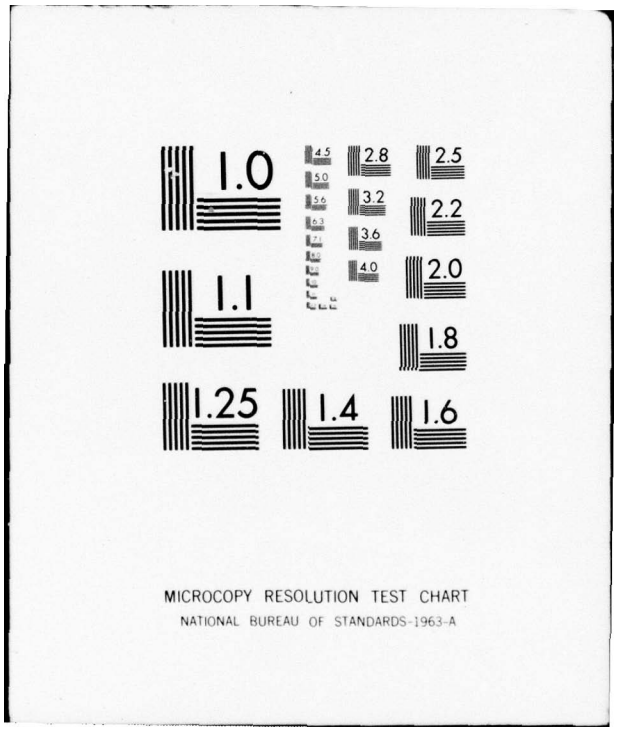
AD-A039 720 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 17/2
MODIFICATION OF THE AN/TGC-26 MULTIPLE ADDRESS PROCESSING UNIT --ETC(U)
MAR 77 R G SHIVELY

UNCLASSIFIED AFIT/GE/EE/77-3 NL

2 OF 2
AD A039720



END
DATE FILMED
6-77


1.0
1.1
1.25
4.5
5.0
5.6
6.3
7.1
8.0
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.8
1.4
1.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

TIMING TABLE-- ORIGINAL CIRCUIT (ASYNC).

TIME	BIT2	BIT3	BIT4	SDET	RESOL	BLIND	DREAD	SDIST	DETCO	READ	SRSP	SRCR	OUTSP	OUTCR	COUNT	RESET
148	1	1	0	1	0	0	1	1	0	0	0	1	1	1	0	0
151	1	1	0	1	0	0	0	1	0	0	0	1	1	1	0	0
156	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	0
158	1	1	0	1	0	1	0	1	0	1	0	0	1	1	0	0
162	1	1	0	1	0	1	1	1	0	0	0	0	1	1	0	0
165	1	1	0	1	0	1	0	1	0	0	0	0	1	1	0	0
168	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	0
171	0	1	1	1	0	1	0	1	0	1	0	0	1	1	0	0
172	0	1	1	1	0	1	1	1	0	0	0	0	1	1	0	0
173	0	1	1	1	0	1	0	1	0	0	0	0	1	1	0	0
175	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	0

$END
NO FATAL ERRORS

BEST AVAILABLE COPY

Fig. 25. Original Circuit DLS Program (Sheet 3 of 3)

```
**************************
* PROGRAM DLS            *
* IS NOW BEING EXECUTED  *
**************************


$DLS     MODIFICATION 1  (ASYNC).                                        00010
$PATCH
       INPUT BIT1  BIT2  BIT3  BIT4  BIT5  SDET  RESDL RESDT DREAD READ
       PRINT BIT0  BIT1  BIT4  SPLT  SPEC  OUTEG OUTLT COUNT RESDT
IBIT1  INV   BIT1
IBIT2  INV   BIT2
IBIT3  INV   BIT3
IBIT4  INV   BIT4
IBIT5  INV   BIT5
I1     INV   PRSDL
I2     INV   SDET
SDIST  INV   DR2
FG     AND   IBIT1 IBIT2 BIT3  IBIT4 IBIT5 DREAD
LT     AND   IBIT1 IBIT2 IBIT3 IBIT4 IBIT5 DREAD
SP     AND   BIT1  BIT2  IBIT3 BIT4  BIT5  DREAD
CR     AND   BIT1  BIT2  BIT3  IBIT4 BIT5  DREAD
LF     AND   BIT1  IBIT2 BIT3  BIT4  BIT5  DREAD
A1     AND   00    FG
A2     AND   04    FG    03
A3     AND   COUNT FG    BLIND
A4     AND   01    LT    02
A5     AND   COUNT LT    BLIND
A6     AND   COUNT SP    BLIND
A7     AND   COUNT 02    05
A8     AND   0F    LF
A9     AND   RESDT I1
A10    AND   NDETC I2
LOCK   AND   NDETC 05
OUTLT  NAND  OUTLT READ
OUTEG  NAND  SPEC  READ
OR1    02    A5    A6    A8
OR2    02    A3    A6    A5    A8
00     JKFF  01    01    A1                01          RESDT
02     JKFF  03    02    A4                03          RESDT
04     JKFF  COUNT 04    A2                COUNT       RESDT
NDETC  JKFF  DETCO NDETC DR2               DETCO       A9
06     JKFF  05    05    A7                05          A8
SPLT   RSFFU A10   DR1
SPEC   RSFFU A10   A8
BLIND  RSFFU DETCO A10
$MODE ASYNC
$INITIAL CONDITIONS
SDIST 1
BLIND 1
00    1
02    1
04    1
06    1
LOCK  1
NDETC 1
OUTEG 1
OUTLT 1
$RUN        10
```

BEST AVAILABLE COPY

Fig. 26. Modification 1 DLS Program (Sheet 1 of 3)

BEST AVAILABLE COPY

Fig. 26. Modification 1 DLS Program
(Sheet 2 of 3)

TIMING TABLE-- MODIFICATION 1 (ASYNC).

TIME	BIT2	BIT3	BIT4	SDET	RESCL	BLIND	DP_AD	SDIST	DETCO	LOCK	READ	SR_T	SRFG	OUTFG	OUTLT	COUNT	RESET
150	1	1	0	1	0	1	1	1	0	0	0	0	0	1	1	1	1
152	1	1	0	1	0	1	0	1	0	0	0	0	0	1	1	1	1
155	1	1	1	0	0	1	0	1	0	0	0	0	0	1	1	1	1
158	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	1
163	0	1	1	1	0	1	1	0	0	0	0	1	0	1	1	1	1
169	0	1	1	1	0	0	0	1	1	0	0	1	0	1	1	1	1
171	1	1	1	0	0	0	0	1	1	0	0	1	0	1	1	1	1
176	1	1	1	1	1	0	0	1	0	1	1	1	0	1	0	1	1
179	1	1	1	1	0	0	1	1	0	1	0	1	0	1	1	1	1
180	1	1	1	1	0	0	0	1	0	1	0	1	0	1	1	1	1
184	1	1	1	0	0	1	0	1	0	1	0	0	0	1	1	1	1
187	0	1	0	1	0	1	0	1	0	1	1	0	0	1	1	1	1
192	0	1	0	1	0	1	1	0	0	1	0	0	1	1	1	1	1
193	0	1	0	1	0	0	0	1	1	0	0	0	1	1	1	1	1
200	1	1	1	0	0	0	0	1	1	0	0	0	1	1	1	1	1
205	1	1	1	1	1	0	0	1	0	1	1	0	1	0	1	1	1
203	1	1	1	1	0	0	1	1	0	1	0	0	1	1	1	1	1
207	1	1	1	1	0	0	0	1	0	1	0	0	1	1	1	1	1
213	1	1	1	0	0	1	0	1	0	1	0	0	0	1	1	1	1

$END
NO FATAL ERRORS

BEST AVAILABLE COPY

Fig. 26. Modification 1 DLS Program
(Sheet 3 of 3)

BEST AVAILABLE COPY

Fig. 27. Modification 2 DLS Program
(Sheet 1 of 4)

```
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SPSP  NAND  HALT  4    0
A    0 NAND  N10   SPSP
*****PRECEDING LINES PRODUCED BY GENDEK*****
ONE    SWTCH
ZERO   SWTCH
$MODE ASYNC
$INITIAL CONDITIONS
SDIST 1
BLIND 1
SRCR  1
OUTSP 1
A4    1
ONE   1
ZERO  0
LOCK  1
NDETC 1
OUTFG 1
OUTLT 1
$RUN          10
```

BEST AVAILABLE COPY

Fig. 27. Modification 2 DLS Program
(Sheet 2 of 4)

BEST AVAILABLE COPY

TIMING TABLE-- MODIFICATION 2 (ASYNC).

TIME	BIT3	BIT2	BIT1	BIT0	SOET	RESCL	BLIVD	DELAY	SDIST	DETCD	LOCK	READ	SOFS	SRLT	SRSP	OUTFG	OUTLT	OUTSP	COUNT	RESET	HALT
0	1	1	1	1	1	1	1	0	1	0	1	0	0	0	0	1	1	1	0	0	1
5	1	1	1	1	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
9	1	1	1	1	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
11	1	1	1	1	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
13	1	1	1	1	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
17	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
21	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
25	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
31	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
35	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
39	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
41	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
43	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
47	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
51	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
55	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
61	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
65	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
69	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
71	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
73	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
77	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
81	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
85	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
91	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
95	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
99	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
101	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
103	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
107	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
111	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
117	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
124	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
126	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
131	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
134	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
135	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
141	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
145	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
151	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
153	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
160	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
165	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
163	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
169	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
175	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
179	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
185	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
192	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
194	1	1	1	0	1	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
199	1	1	1	0	0	1	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1

Fig. 27. Modification 2 DLS Program
(Sheet 3 of 4)

BEST AVAILABLE COPY

TIMING TABLE-- MODIFICATION 2 (ASYNC).

TIME	BIT2	BIT3	BIT4	SDET	RESOL	BLIND	DREAD	SDIST	DETCD	LOCK	READ	SRFS	SRLT	SRSP	OUTFG	OUTLT	OUTSP	COUNT	RESCT	HALT
202	1	1	1	1	0	0	1	1	0	1	0	0	0	1	1	1	1	1	1	1
203	1	1	1	1	0	0	0	1	0	1	0	0	0	1	1	1	1	1	1	1
208	1	1	1	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
212	1	0	1	1	0	1	0	1	0	1	1	0	0	0	1	1	1	1	1	1
218	1	0	1	1	0	1	1	0	0	1	0	0	1	0	1	1	1	1	1	1
225	1	0	1	1	0	0	0	1	1	0	0	0	1	0	1	1	1	1	1	1
230	1	1	1	1	1	0	0	1	0	1	1	0	0	1	1	1	0	1	1	0
233	1	1	1	1	0	0	1	1	0	1	0	0	0	1	1	1	1	1	1	1
234	1	1	1	1	0	0	0	1	0	1	0	0	0	1	1	1	1	1	1	1
239	1	1	1	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
243	1	1	0	1	0	1	0	1	0	1	1	0	0	0	1	1	1	1	1	1
247	1	1	0	1	0	1	1	1	0	1	0	0	0	0	1	1	1	1	1	1
253	1	1	0	1	0	1	0	1	0	0	0	0	0	0	1	1	1	1	1	1
257	1	1	1	0	0	1	0	1	0	0	0	0	0	0	1	1	1	1	1	1
258	1	1	1	0	0	1	0	1	0	0	0	0	0	0	1	1	1	1	1	1
262	1	1	0	1	0	1	0	1	0	0	1	0	0	0	1	1	1	1	1	1
264	1	1	0	1	0	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1
266	1	1	0	1	0	1	0	1	0	0	0	0	0	0	1	1	1	1	1	1
270	1	1	1	0	0	1	0	1	0	0	0	0	0	0	1	1	1	1	1	1
274	0	1	1	1	0	1	0	1	0	0	1	0	0	0	1	1	1	1	1	1
280	0	1	1	1	0	1	1	0	0	0	0	0	1	0	1	1	1	1	1	1
287	0	1	1	1	0	0	0	1	1	0	0	0	1	0	1	1	1	1	1	1
289	1	1	1	0	0	0	0	1	1	0	0	0	1	0	1	1	1	1	1	1
294	1	1	1	1	1	0	0	1	0	1	1	0	1	0	1	0	1	1	1	1
297	1	1	1	1	0	0	1	1	0	1	0	0	1	0	1	1	1	1	1	1
298	1	1	1	1	0	0	0	1	0	1	0	0	1	0	1	1	1	1	1	1
304	1	1	1	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1
308	0	1	0	1	0	1	0	1	0	1	1	0	0	0	1	1	1	1	1	1
313	0	1	0	1	0	1	1	0	0	1	0	1	0	0	1	1	1	1	1	1
320	0	1	0	1	0	0	0	1	1	0	0	1	0	0	1	1	1	1	1	1
322	1	1	1	0	0	0	0	1	1	0	0	1	0	0	1	1	1	1	1	1
327	1	1	1	1	1	0	0	1	0	1	1	1	0	0	0	1	1	1	1	1
330	1	1	1	1	0	0	1	1	0	1	0	1	0	0	1	1	1	1	1	1
331	1	1	1	1	0	0	0	1	0	1	0	1	0	0	1	1	1	1	1	1
336	1	1	1	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1

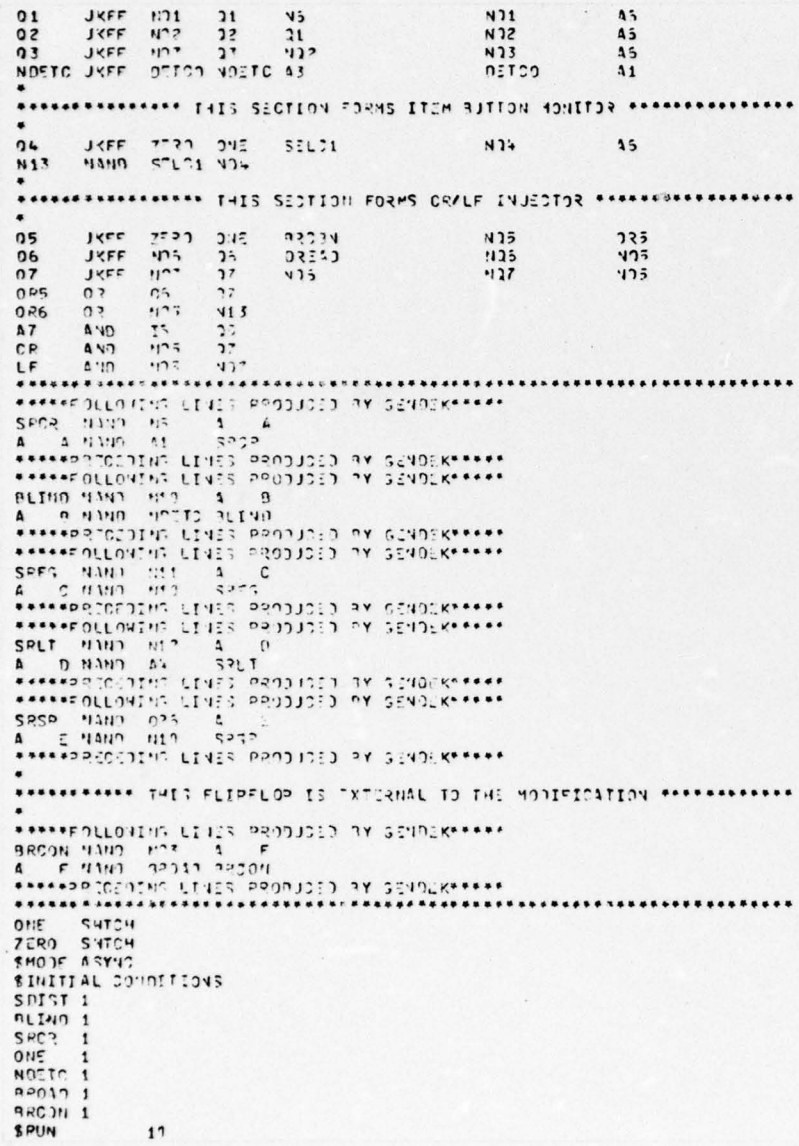
$END
NO FATAL ERRORS

Fig. 27. Modifcation 2 DLS Program
(Sheet 4 of 4)

BEST AVAILABLE COPY

```
*****************************
* PROGRAM DLS               *
* IS NOW BEING ENTERED      *
*****************************


$DLS    MODIFICATION 3 (ASM80).                                    00010
$PATCH
        INPUT BIT1  BIT2  BIT3  BIT4  BIT5  SORT  RESE  READ READ  START
        INPUT SELC1 SRCH4
        PRINT BIT1  BIT2  BIT3  BIT4  SORT  RESE  N-END SORT DETCD CLOCK
        PRINT SRCH  OUT1  OUT2  OUT3  OUT4  OUT5  START SELC1 SRCH4

*****************************  THIS SECTION FORMS CODE DETECTOR  *****************************
*
IBIT1  INV   BIT1
IBIT2  INV   BIT2
IBIT3  INV   BIT3
IBIT4  INV   BIT4
IBIT5  INV   BIT5
N1     NAND  IBIT1 IBIT2 IBIT3 IBIT4 IBIT5 N-END N1    DREAD
N2     NAND  IBIT1 IBIT2 IBIT3 IBIT4 IBIT5 N-END N11   DREAD
N3     NAND  BIT1  IBIT2 IBIT3 BIT4  BIT5  N-END N1    DREAD
N4     NAND  IBIT1 BIT2  BIT3  IBIT4 BIT5  N-END SRCH  DREAD
N5     NAND  BIT1  BIT2  BIT3  IBIT4 BIT5  N-END N13   DREAD
*****************************************************************************************
I6     INV   SORT
I7     INV   SRCH
SORST  INV   A7
I8     INV   RESET
I9     INV   N1
I10    INV   A5
RESET  INV   SORST
A1     AND   IP      IP
LOCK   AND   IP      N7      VDETC
A3     AND   N13     N10     N13
A4     AND   N8      N9      N3
BROAD  AND   N8      N9      N1      V+
A5     AND   RESET   SRCH    N4
A6     AND   N11     N4      V+
N6     NAND  N1      N2      A5
N7     NAND  N1      N2      A5
N8     NAND  SRCH    N4      N13
N9     NAND  N13     N2      N13
N10    NAND  A7      N2      VDETC
N11    NAND  I7      N3      N13
N12    NAND  I10     N3      N13
*
*****************************  THIS SECTION FORMS OUTPUT ENCODER  *****************************
*
OUT1   NAND  OP1     READ
OUT2   NAND  OP2     READ
OUT3   NAND  OP3     READ
OUT4   NAND  OP4     READ
OUT5   NAND  OP1     SELF
OP1    OR    SRCH    LP
OP2    OR    OP1     SRCH
OP3    OR    SELF    SRCH
OP4    OR    OP1     SP
*
*****************************  THIS SECTION FORMS CODE COUNTER  *****************************
```

Fig. 28. Modification 3 DLS Program (Sheet 1 of 6)



```
Q1    JKFF  N21   J1    N5                N21         A5
Q2    JKFF  N22   J2    J1                N22         A5
Q3    JKFF  N23   J3    N23               N23         A5
NDETC JKFF  DETCO NDETC A3                DETCO       A1
*
*************** THIS SECTION FORMS ITEM BUTTON MONITOR ***************
*
Q4    JKFF  ZERO  ONE   SEL21             N14         A5
N13   NAND  SEL21 N14
*
***************** THIS SECTION FORMS CR/LF INJECTOR ******************
*
Q5    JKFF  ZERO  ONE   BRCRN             N15         CR5
Q6    JKFF  N25   J5    CREAD             N15         N25
Q7    JKFF  N27   J7    N15               N17         N25
OR5   OR    Q5    J7
OR6   OR    N25   N13
A7    AND   I5    J2
CR    AND   N25   J7
LF    AND   N25   N27
*****************************************************************
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SRCR  NAND  N5    A   A
A   A NAND  A1    SRCR
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
BLIND NAND  N10   A   B
A   B NAND  NDETC BLIND
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SREG  NAND  N11   A   C
A   C NAND  N12   SREG
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SPLT  NAND  N13   A   D
A   D NAND  A4    SPLT
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SRSP  NAND  CR5   A   E
A   E NAND  N10   SRSP
*****PRECEDING LINES PRODUCED BY GENDEK*****
*
********** THIS FLIPFLOP IS EXTERNAL TO THE MODIFICATION **********
*
*****FOLLOWING LINES PRODUCED BY GENDEK*****
BRCRN NAND  N27   A   F
A   F NAND  BRDIV BRCRN
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****************************************************************
ONE   SWTCH
ZERO  SWTCH
$MODE ASYNC
$INITIAL CONDITIONS
SDIST 1
BLIND 1
SRCR  1
ONE   1
NDETC 1
BRDIV 1
BRCRN 1
$RUN        11
```

BEST AVAILABLE COPY

Fig. 28. Modification 3 DLS Program (Sheet 2 of 6)

BEST AVAILABLE COPY

TIMING TABLE-- MODIFICATION 3 (ASYNC).

TIME	BIT2	BIT3	BIT4	SDET	RESCL	BLIND	ORED	SDIST	DETCD	DISK	READ	SRCR	OUT1	OUT2	OUT3	OUT4	OUT5	START	SELC1	BRDBN
0	1	1	1	0	1	1	0	1	0	0	0	1	0	0	0	0	0	1	0	0
6	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
11	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
12	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
13	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
19	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
23	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
25	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
31	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
35	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
41	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
42	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
43	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
48	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
53	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
57	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
61	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
66	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
71	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
72	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
73	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
78	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
83	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
85	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
92	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
97	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
102	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
103	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
104	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
107	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
114	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
122	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
130	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
133	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
139	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
141	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
142	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
151	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
156	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
164	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
172	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
175	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
176	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
184	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
187	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
189	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
191	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
194	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
197	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
202	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0
204	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0

Fig. 28. Modification 3 DLS Program
(Sheet 3 of 6)

BEST AVAILABLE COPY

```
TIMING TABLE--   MODIFICATION 3  (ASYNC).

 TIME  BIT2  BIT3  BIT4  SDET  RESCL  BLIND  DREAD  SDIST  DETCD  -DCK  READ  SRCR  OUT1  OUT2  OUT3  OUT4  OUT5  START  SELC1  SRCBN
  207     1     1     1     1      1      0      0      1      0      1     1     0     1     1     1     1     1      0      0      0
  210     1     1     1     1      1      0      0      1      1      1     1     0     1     1     1     1     1      0      0      0
  216     1     1     1     1      0      0      0      1      1      1     1     0     1     1     1     1     1      0      0      0
  222     1     1     1     0      0      0      0      1      1      1     1     0     1     1     1     1     1      0      0      0
  227     1     1     1     0      0      1      0      1      1      1     1     0     1     1     1     1     1      0      0      0
  235     1     1     1     0      0      1      1      1      1      1     1     0     1     1     1     1     1      0      0      0
  243     1     1     1     0      0      1      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  246     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  254     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      1      0
  257     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  259     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  265     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  270     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  274     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  286     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  293     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  297     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  299     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  307     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  312     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  317     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  319     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  324     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  327     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  331     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  333     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  343     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  351     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  353     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  362     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  370     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  373     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  374     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  381     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  385     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  391     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  407     1     1     1     0      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  410     1     1     1     1      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  415     1     1     1     1      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  416     1     1     1     1      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  419     1     1     1     1      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
  427     1     1     1     1      0      0      1      1      0      1     1     0     1     1     1     1     1      0      0      0
*RECYCL*
```

Fig. 28. Modification 3 DLS Program
(Sheet 4 of 6)

BEST AVAILABLE COPY

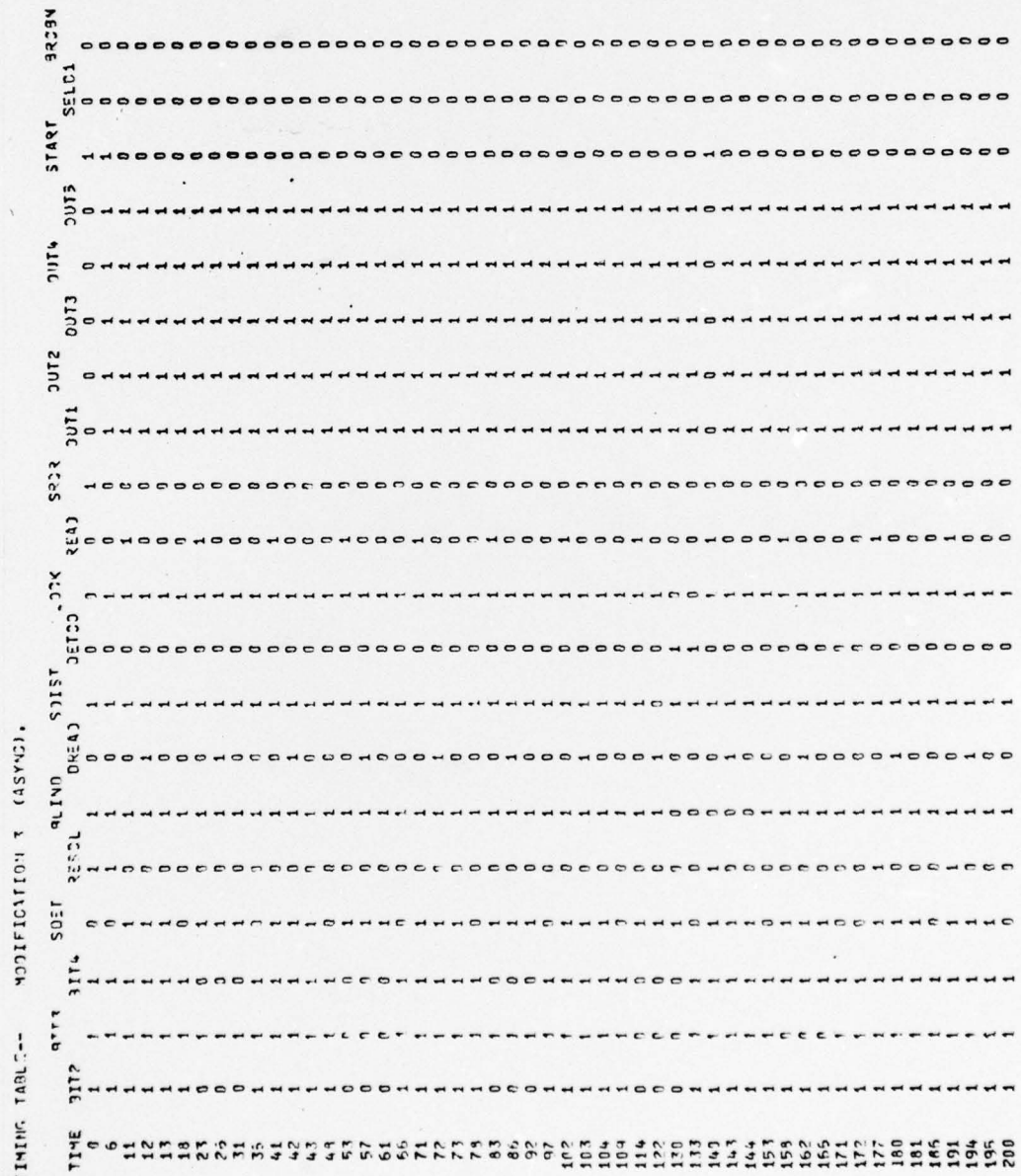


Fig. 28. Modification 3 DLS Program
(Sheet 5 of 6)

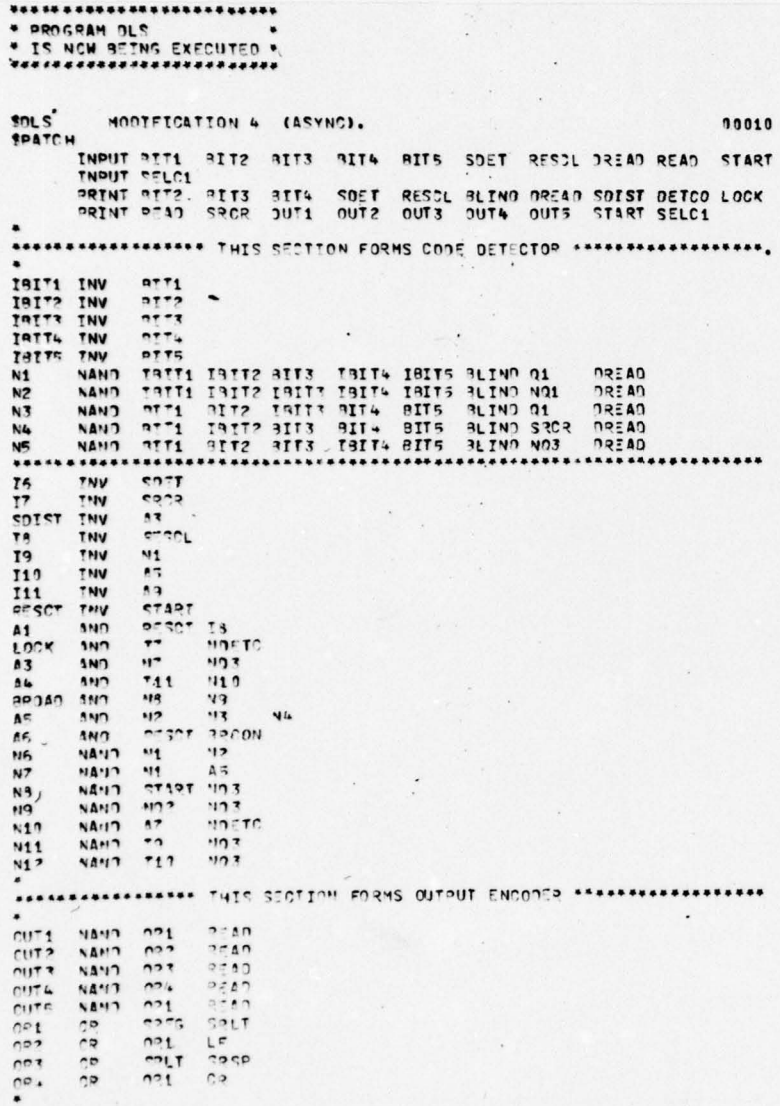
TIMING TABLE-- MODIFICATION 3 (ASYNC).

TIME	BIT3		BIT4	SDET	RESOL	BLIND	DREAD	SDIST	DETCO	LOCK	READ	SRCR	OUT1	OUT2	OUT3	OUT4	OUT5	START	SELC1	BRDBN
205	1	1	1	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
208	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
209	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
214	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
219	1	0	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
223	1	0	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
227	1	0	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
232	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
237	1	1	1	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
240	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
241	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
245	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
251	1	0	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
255	1	0	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
259	1	0	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
264	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
269	1	1	1	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
272	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
273	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
278	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
283	1	1	0	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
284	1	1	0	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
285	1	1	0	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
290	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
295	1	1	0	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
296	1	1	0	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
297	1	1	0	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
302	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
307	0	1	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
308	0	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
309	0	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
314	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
319	1	1	1	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
322	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
323	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
328	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
333	0	1	0	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
335	0	1	0	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
340	0	1	0	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
345	1	1	1	0	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0	0
350	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
353	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0
354	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0
359	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0	0

$END

NO FATAL ERRORS

Fig. 28. Modification 3 DLS Program (Sheet 6 of 6)



```
************************
* PROGRAM DLS          *
* IS NOW BEING EXECUTED *
************************


$DLS     MODIFICATION 4  (ASYNC).                                    00010
$PATCH
      INPUT BIT1  BIT2  BIT3  BIT4  BIT5  SDET  RESDL DREAD READ  START
      INPUT SELC1
      PRINT BIT2  BIT3  BIT4  SDET  RESDL BLIND DREAD SDIST DETCO LOCK
      PRINT READ  SRCR  OUT1  OUT2  OUT3  OUT4  OUT5  START SELC1
*
****************** THIS SECTION FORMS CODE DETECTOR ******************.
*
IBIT1 INV   BIT1
IBIT2 INV   BIT2
IBIT3 INV   BIT3
IBIT4 INV   BIT4
IBIT5 INV   BIT5
N1    NAND  IBIT1 IBIT2 BIT3  IBIT4 IBIT5 BLIND Q1    DREAD
N2    NAND  IBIT1 IBIT2 IBIT3 IBIT4 IBIT5 BLIND NQ1   DREAD
N3    NAND  BIT1  BIT2  IBIT3 BIT4  BIT5  BLIND Q1    DREAD
N4    NAND  BIT1  IBIT2 BIT3  BIT4  BIT5  BLIND SRCR  DREAD
N5    NAND  BIT1  BIT2  BIT3  IBIT4 BIT5  BLIND NQ3   DREAD
**************************************************************************
I6    INV   SDET
I7    INV   SRCR
SDIST INV   A3
I8    INV   RESDL
I9    INV   N1
I10   INV   A5
I11   INV   A3
RESET INV   START
A1    AND   RESET I8
LOCK  AND   I7    NDETC
A3    AND   N7    NQ3
A4    AND   I11   N10
READ  AND   N8    N9
A5    AND   N2    N3    N4
A6    AND   RESET READON
N6    NAND  N1    N2
N7    NAND  N1    A5
N8    NAND  START NQ3
N9    NAND  NQ3   NQ3
N10   NAND  A7    NDETC
N11   NAND  I9    NQ3
N12   NAND  I11   NQ3
*
****************** THIS SECTION FORMS OUTPUT ENCODER ******************
*
OUT1  NAND  OR1   READ
OUT2  NAND  OR2   READ
OUT3  NAND  OR3   READ
OUT4  NAND  OR4   READ
OUT5  NAND  OR1   READ
OR1   OR    READ  SELT
OR2   OR    OR1   LE
OR3   OR    SELT  SRCR
OR4   OR    OR1   OR
*
```

Fig. 29. Modification 4 DLS Program (Sheet 1 of 4)

```
****************** THIS SECTION FORMS CODE COUNTER ********************
*
Q1    JKFF  NQ1   Q1    N6                NQ1         A6
Q2    JKFF  NQ2   Q2    Q1                NQ2         A6
Q3    JKFF  NQ3   Q3    NQ2               NQ3         A5
NDETC JKFF  DETCO NDETC A3                DETCO       A1
*
*************** THIS SECTION FORMS ITEM BUTTON MONITOR ***************
*
Q4    JKFF  ZERO  ONE   SELC1             NQ4         A6
Q8    JKFF  NQ8   Q8    SELC1             NQ8         A6
Q9    JKFF  NQ9   Q9    NQ8               NQ9         A6
Q10   JKFF  NQ10  Q10   NQ9               NQ10        A6
A8    AND   Q8    Q9    NQ10  A3
A9    AND   SELC1 NQ4
*
***************** THIS SECTION FORMS CR/LF INJECTOR ******************
*
Q5    JKFF  ZERO  ONE   A8                NQ5         N14
Q6    JKFF  NQ6   Q6    QREAD             NQ6         NQ5
Q7    JKFF  NQ7   Q7    NQ5               NQ7         NQ5
N14   NAND  NQ7   QREAD
N13   NAND  A9    Q5
A7    AND   T6    Q5
CR    AND   NQ6   Q7
LF    AND   NQ5   NQ7
*********************************************************************
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SRCR  NAND  N5    A   A
A   A NAND  A1    SRCR
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
BLIND NAND  N10   A   B
A   B NAND  NDETC BLIND
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SREG  NAND  N11   A   C
A   C NAND  N11   SREG
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SRLT  NAND  N12   A   D
A   D NAND  A4    SRLT
*****PRECEDING LINES PRODUCED BY GENDEK*****
*****FOLLOWING LINES PRODUCED BY GENDEK*****
SRSR  NAND  N13   A   E
A   E NAND  N11   SRSR
*****PRECEDING LINES PRODUCED BY GENDEK*****
*
********** THIS FLIPFLOP IS EXTERNAL TO THE MODIFICATION **********
*
*****FOLLOWING LINES PRODUCED BY GENDEK*****
PRCON NAND  NQ7   A   F
A   F NAND  PREAD PRCON
*****PRECEDING LINES PRODUCED BY GENDEK*****
*********************************************************************
```

```
ONE    SWTCH
ZERO   SWTCH
$MODE  ASYNC
$INITIAL CONDITIONS
SQIST  1
BLIND  1
SRCR   1
ONE    1
NDETC  1
PREAD  1
PRCON  1
$RUN        10
```

Fig. 29. Modification 4 DLS Program (Sheet 2 of 4)

TIMING TABLE-- MODIFICATION 4 (ASYNC).

TIME	BIT2	BIT3	BIT4	SDET	RESCL	BLIND	OREAD	SDIST	DETCO	LOCK	READ	SRCR	OUT1	OUT2	OUT3	OUT4	OUT5	START	SELC1
0	1	1	1	0	1	1	0	1	0	0	0	1	0	0	0	0	0	1	0
6	1	1	1	0	1	1	0	1	0	1	0	0	1	1	1	1	1	1	0
11	1	1	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
12	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0
13	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
18	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
23	0	1	0	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
26	0	1	0	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0
31	0	1	0	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
36	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
41	1	1	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
42	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0
43	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
48	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
53	0	0	0	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
57	0	0	0	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0
61	0	0	0	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
66	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
71	1	1	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
72	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0
73	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
78	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
83	0	1	0	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
86	0	1	0	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0
92	0	1	0	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
97	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
102	1	1	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
103	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	1	0	0
104	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
109	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
114	0	0	0	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
122	0	0	0	1	0	1	1	0	1	1	0	0	1	1	1	1	1	0	0
130	0	0	0	1	0	0	0	1	1	0	0	0	1	1	1	1	1	0	0
133	1	1	1	0	0	0	0	1	1	0	0	0	1	1	1	1	1	0	0
138	1	1	1	1	0	0	1	1	0	1	1	0	0	0	0	0	0	0	1
141	1	1	1	1	0	0	0	1	0	1	0	0	1	1	1	1	1	0	0
142	1	1	1	1	0	0	1	1	0	1	0	0	1	1	1	1	1	0	0
151	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
156	1	0	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	0	0
164	1	0	1	1	0	1	1	0	0	1	0	0	1	1	1	1	1	0	0
172	1	0	1	1	0	0	0	1	0	1	0	0	1	1	1	1	1	0	0
173	1	1	1	0	0	0	0	1	1	0	0	0	1	1	1	1	1	0	0
183	1	1	1	1	1	0	0	1	0	1	1	0	1	1	0	1	1	0	1
186	1	1	1	1	0	0	1	1	0	1	0	0	1	1	1	1	1	0	0
187	1	1	1	1	0	0	0	1	0	1	0	0	1	1	1	1	1	0	0
194	1	1	1	0	0	1	0	1	0	1	0	0	1	1	1	1	1	0	0
199	1	0	1	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	0
207	1	0	1	1	0	1	1	0	0	1	0	0	1	1	1	1	1	0	0
213	1	0	1	1	0	0	0	1	0	1	0	0	1	1	1	1	1	0	0
218	1	1	1	0	0	0	0	1	0	1	0	0	1	1	1	1	1	0	0
226	1	1	1	1	1	0	0	1	0	1	1	0	1	1	0	1	1	0	1

Fig. 29. Modification 4 DLS Program (Sheet 3 of 4)

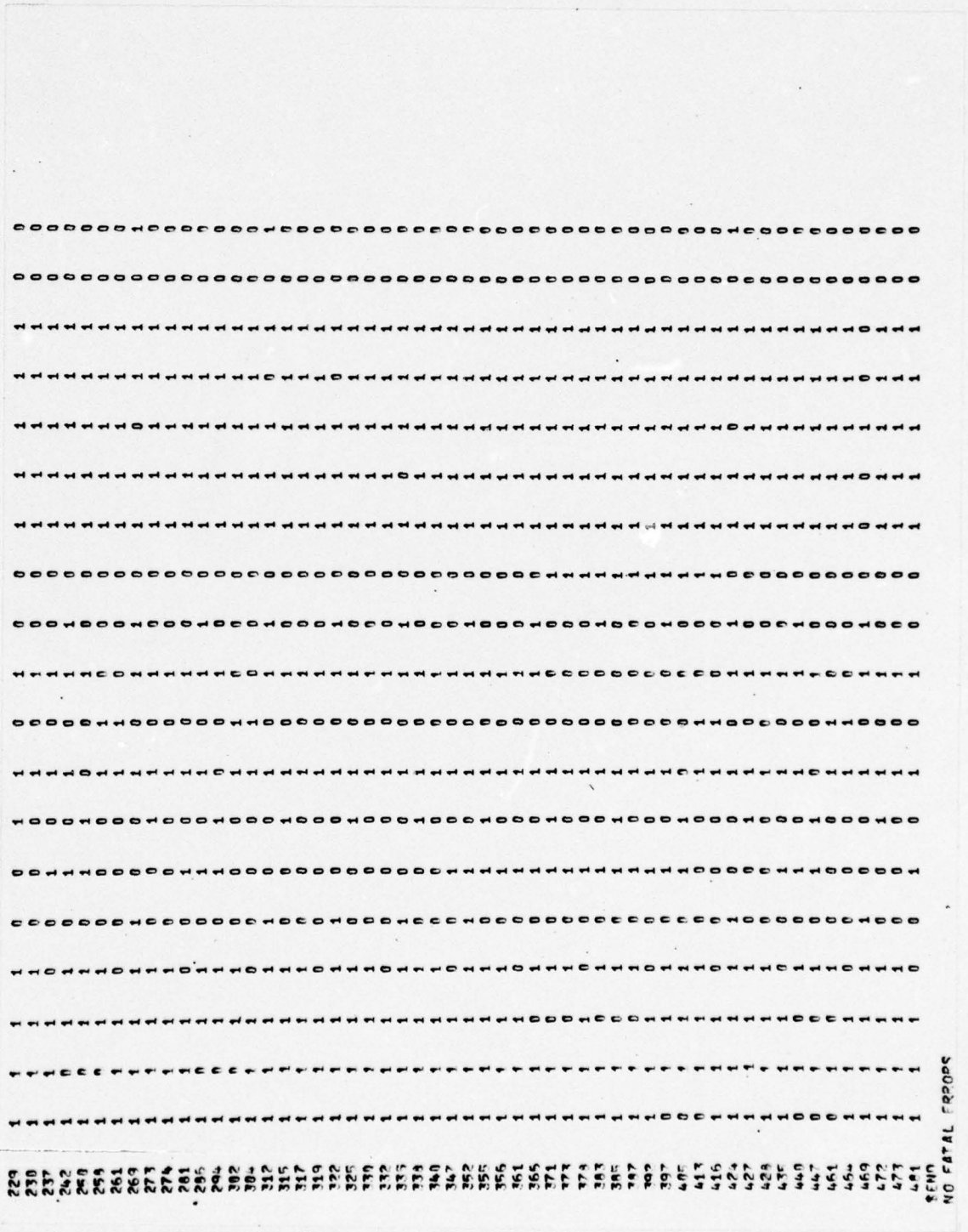
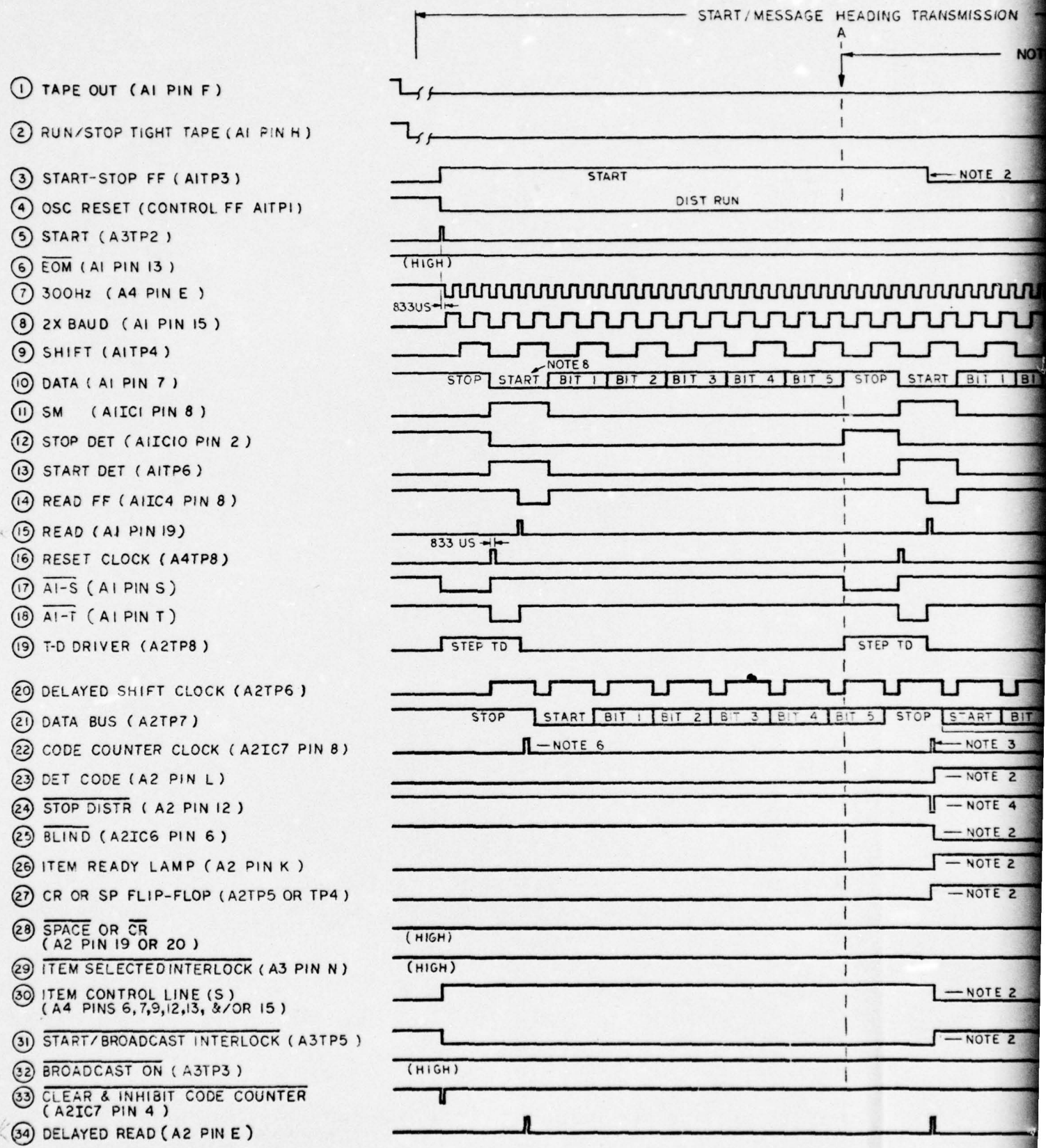


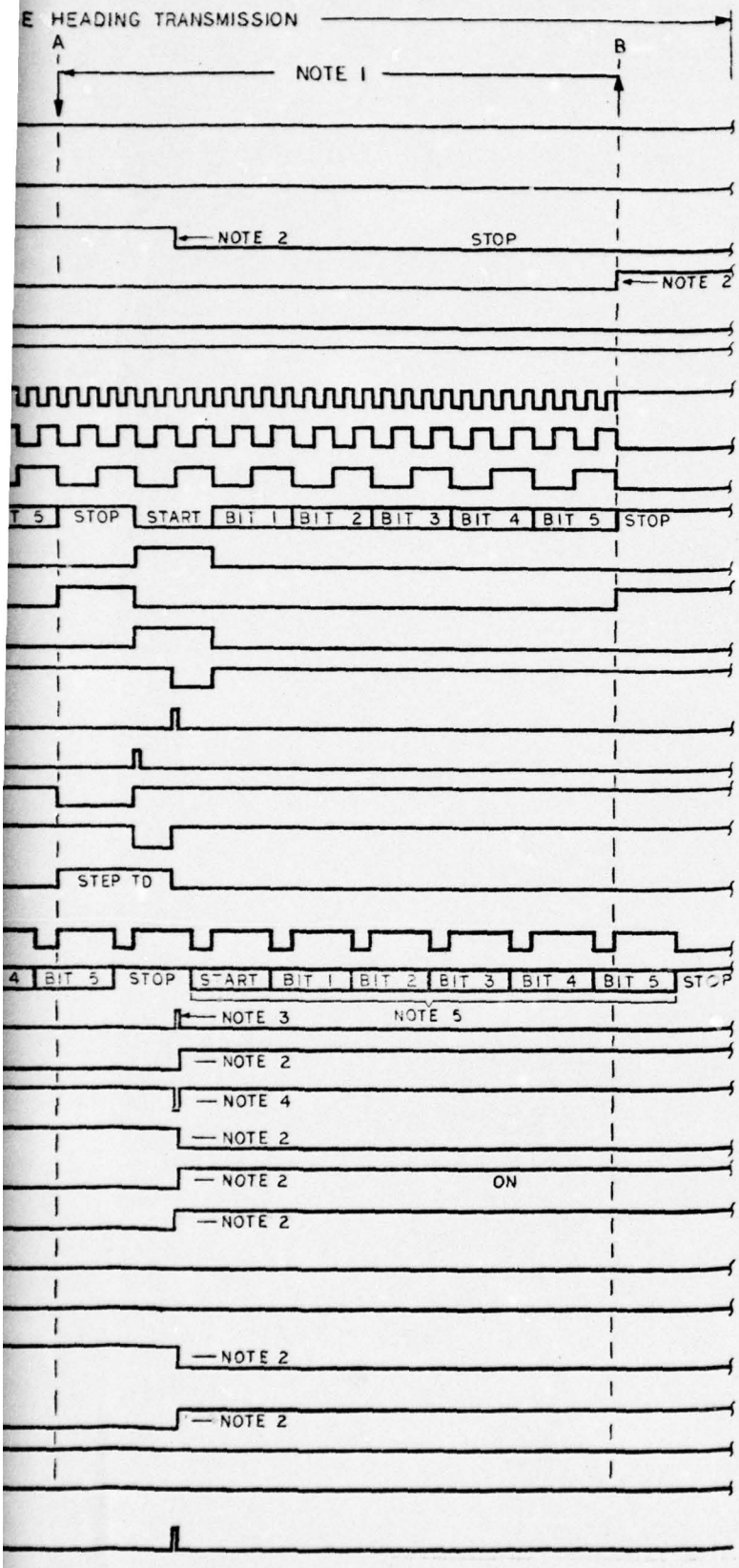
Fig. 29. Modification 4 DLS Program
(Sheet 4 of 4)

APPENDIX B

MAPU Master Timing Diagram

The timing diagram of Fig. 30 shows the timing relationship of the major signals within the MAPU.


START/MESSAGE HEADING TRANSMISSION
A
NOT
1 TAPE OUT (A1 PIN F)
2 RUN/STOP TIGHT TAPE (A1 PIN H)
3 START-STOP FF (A1TP3)
START
NOTE 2
4 OSC RESET (CONTROL FF A1TP1)
DIST RUN
5 START (A3TP2)
6 EOM (A1 PIN 13)
(HIGH)
7 300Hz (A4 PIN E)
833US
8 2X BAUD (A1 PIN 15)
9 SHIFT (A1TP4)
10 DATA (A1 PIN 7)
NOTE 8
STOP START BIT 1 BIT 2 BIT 3 BIT 4 BIT 5 STOP START BIT 1 BI
11 SM (A11C1 PIN 8)
12 STOP DET (A11C10 PIN 2)
13 START DET (A1TP6)
14 READ FF (A1IC4 PIN 8)
15 READ (A1 PIN 19)
833 US
16 RESET CLOCK (A4TP8)
17 A1-S (A1 PIN S)
18 A1-T (A1 PIN T)
19 T-D DRIVER (A2TP8)
STEP TD
STEP TD
20 DELAYED SHIFT CLOCK (A2TP6)
21 DATA BUS (A2TP7)
STOP START BIT 1 BIT 2 BIT 3 BIT 4 BIT 5 STOP START BIT
22 CODE COUNTER CLOCK (A2IC7 PIN 8)
NOTE 6
NOTE 3
23 DET CODE (A2 PIN L)
NOTE 2
24 STOP DISTR (A2 PIN 12)
NOTE 4
25 BLIND (A2IC6 PIN 6)
NOTE 2
26 ITEM READY LAMP (A2 PIN K)
NOTE 2
27 CR OR SP FLIP-FLOP (A2TP5 OR TP4)
NOTE 2
28 SPACE OR CR (A2 PIN 19 OR 20)
(HIGH)
29 ITEM SELECTED INTERLOCK (A3 PIN N)
(HIGH)
30 ITEM CONTROL LINE (S) (A4 PINS 6,7,9,12,13, &/OR 15)
NOTE 2
31 START/BROADCAST INTERLOCK (A3TP5)
NOTE 2
32 BROADCAST ON (A3TP3)
(HIGH)
33 CLEAR & INHIBIT CODE COUNTER (A2IC7 PIN 4)
34 DELAYED READ (A2 PIN E)



NOTES:

1. CHARACTER TRANSMISSION CYCLE (A-B) REPEATS UNTIL LAST END-OF-MESSAGE-HEADING CODE CHARACTER IS DETECTED IN DATA (SHIFT) REGISTER (DET CODE HIGH), AND THEN STOPS AS SHOWN.
2. TRANSITION OCCURS ONLY WHEN LAST CODE CHARACTER HAS BEEN DETECTED IN DATA REGISTER.
3. PULSE OCCURS ONLY WHEN FIRST OR SECOND CODE CHARACTER IS IN DATA REGISTER.
4. PULSE OCCURS ONLY WHEN LAST CODE CHARACTER IS IN DATA REGISTER.
5. DATA BUS REMAINS HIGH (IN STOP-MARK CONDITION) AFTER LAST CODE CHARACTER IS DETECTED IN DATA REGISTER (DET CODE HIGH); LAST CODE CHARACTER NOT TRANSMITTED IN THIS CYCLE.
6. PULSE WILL OCCUR ONLY IF FIRST CHARACTER READ FROM TAPE IS FIRST CODE CHARACTER.
7. CHARACTER TRANSMISSION CYCLE (C-D) REPEATS UNTIL END-OF-ROUTINE-INDICATOR-CODE CHARACTER IS DETECTED (G) IN DATA REGISTER (DET CODE HIGH), AND THEN STOPS AS SHOWN.
8. BIT INTERVAL IS 13.33 MS AT 75 BAUD, 20 MS AT 50 BAUD, OR 26.66 MS AT 37.5 BAUD.
9. READ (15) AND $\overline{\text{SPACE}}$ OR $\overline{\text{CR}}$ (28) PULSES ARE NORMALLY 20 US WIDE; START (5) $\overline{\text{CODE COUNTER CLOCK}}$ (22) $\overline{\text{STOP DIST}}$ (24) $\overline{\text{CLEAR \& INHIBIT CODE COUNTER}}$ (33) (SHEET 1), AND DELAYED READ (34) PULSES ARE NOMINALLY 10 US WIDE. ALL OTHER PULSE WIDTHS SHOWN TO BIT-INTERVAL SCALE.
10. PULSE OCCURS IF FIRST CODE CHARACTER IS READ INTO DATA REGISTER, BUT DOES NOT STEP CODE COUNTER.
11. ALL WAVEFORMS SHOWN ARE FROM POINT OF MEASUREMENT TO SIGNAL GROUND. ALL MEASUREMENT POINTS ARE LOCATED IN PRIMARY CONTROL ASSEMBLY (1A1).
12. LOGIC LEVELS ARE NOMINALLY +4 TO +5 VOLTS (LOGICAL 1 OR HIGH) AND -.7 TO +.5 VOLT (LOGICAL 0 OR LOW).

Fig. 30. MAPU Master Timing Diagram (Ref 1:4-15)

APPENDIX C

MAPU Card Assembly Circuit Diagrams

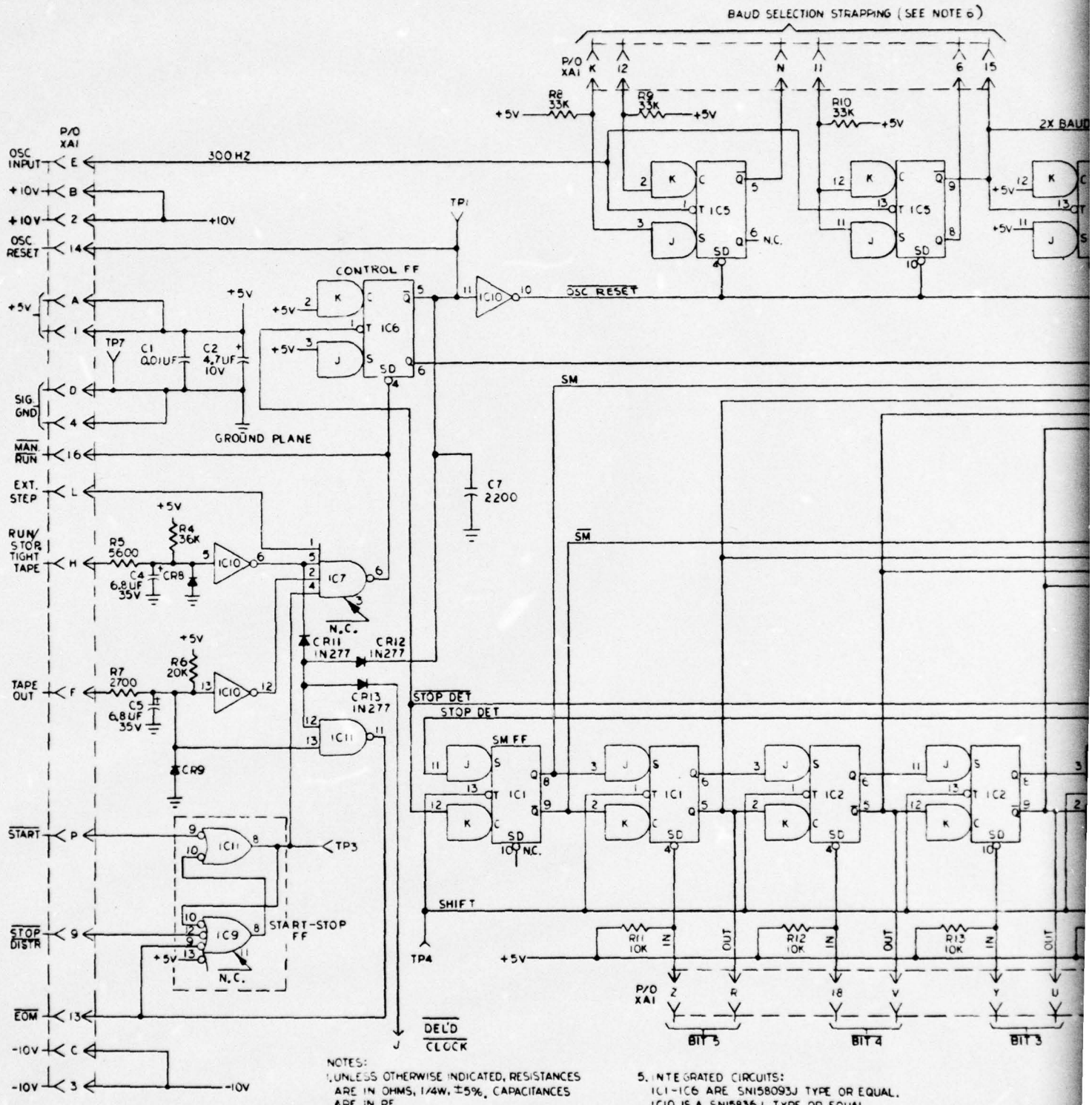
The following appendix contains the schematic diagrams for the four MAPU circuit cards. These diagrams are identified below.

Figure 31: A1 Assembly, character distribution and timing circuits.

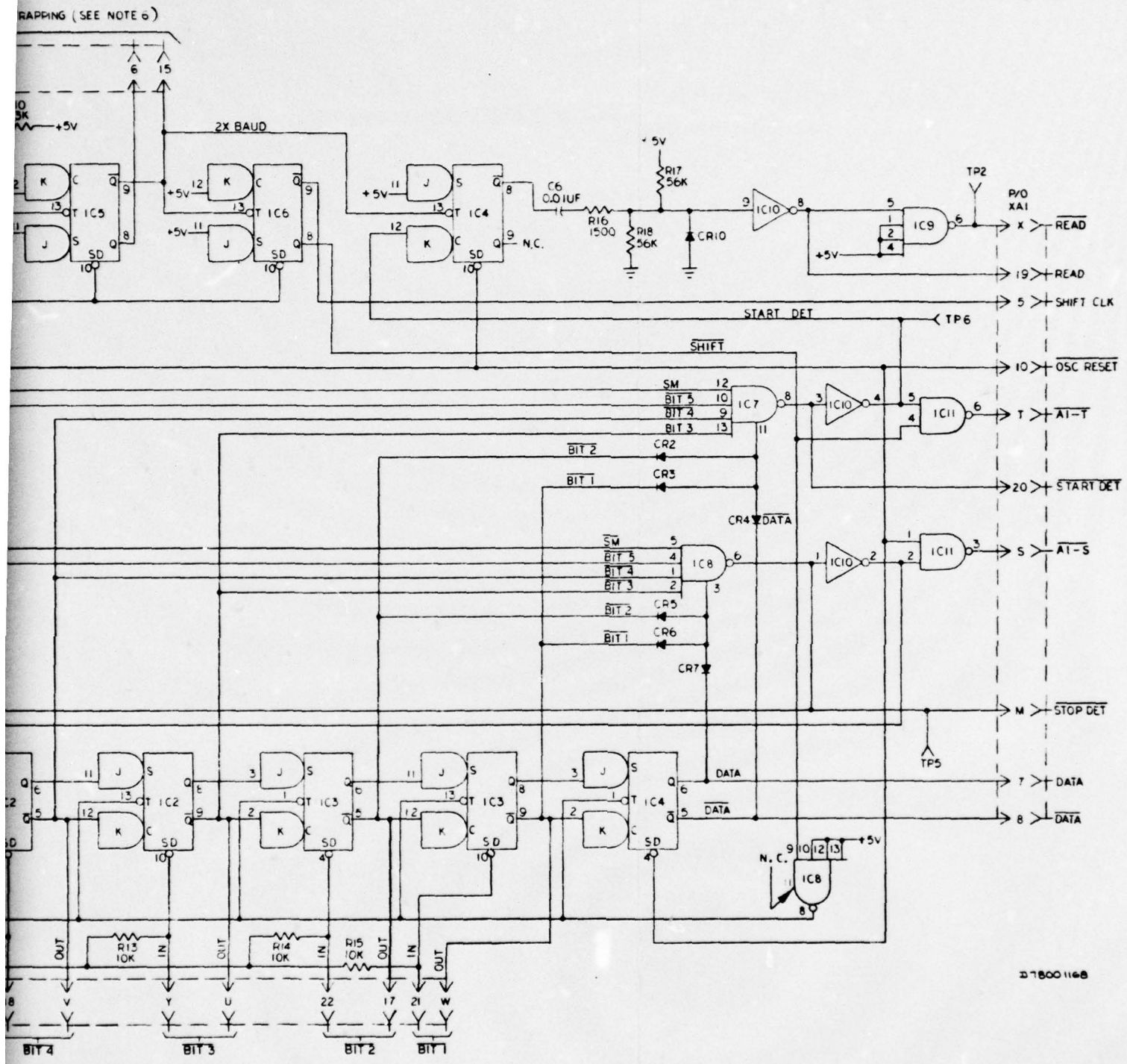
Figure 32: A2 Assembly, code recognition logic circuits.

Figure 33: A3 Assembly, control panel circuits.

Figure 34: A4 Assembly, oscillator and interface circuits.

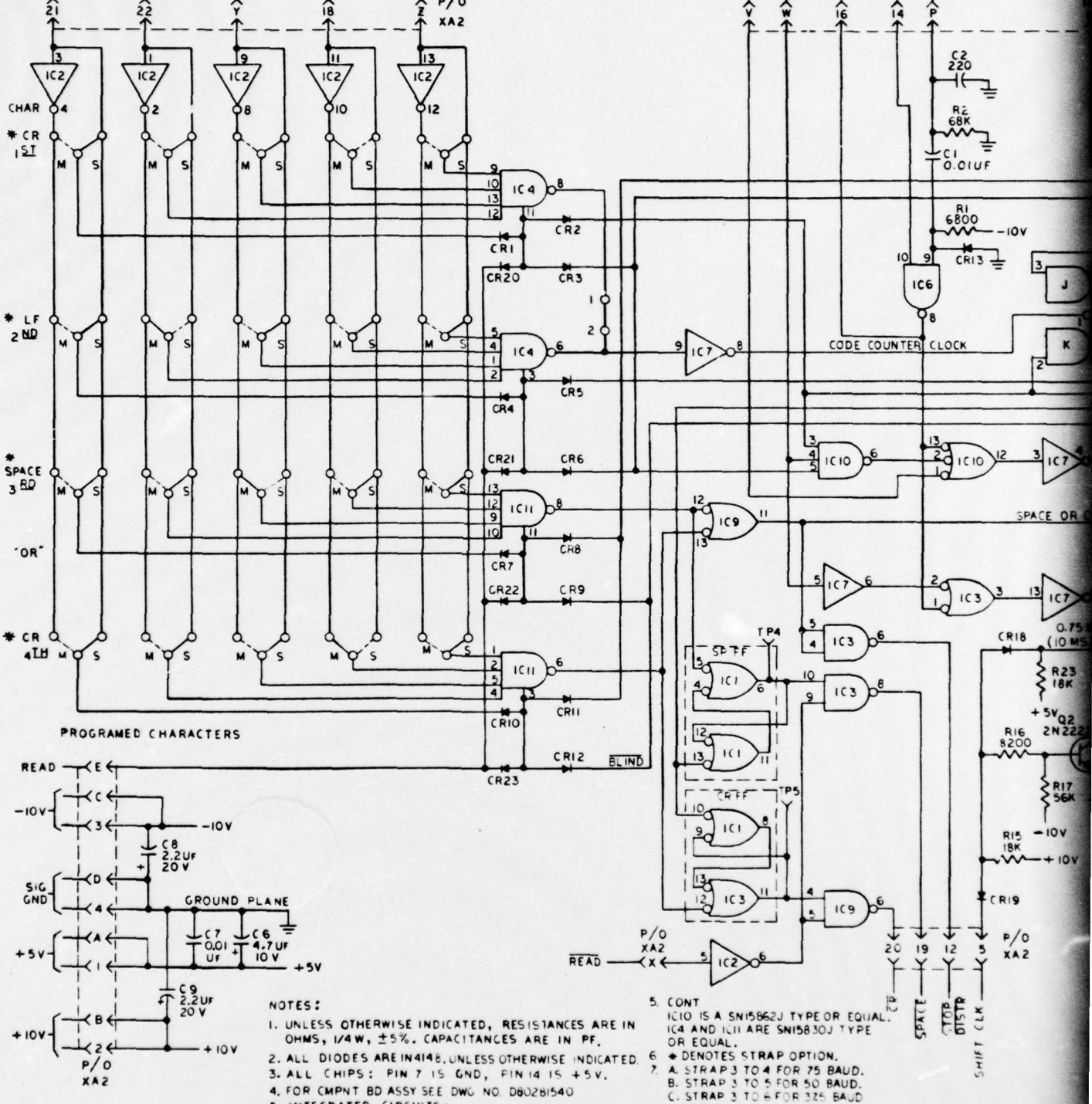


NOTES:
1. UNLESS OTHERWISE INDICATED, RESISTANCES ARE IN OHMS, 1/4W, ±5%. CAPACITANCES ARE IN PF.
2. ALL DIODES ARE IN4148 UNLESS OTHERWISE INDICATED.
3. ALL CHIPS: PIN 7 IS GND
   PIN 14 IS +5V
4. FOR CMPNT BD ASSY SEE DWG NO.D80281530.
5. INTEGRATED CIRCUITS:
   IC1-IC6 ARE SN158093J TYPE OR EQUAL.
   IC10 IS A SN15836J TYPE OR EQUAL.
   IC11 IS A SN15846J TYPE OR EQUAL.
   IC8 AND IC9 ARE SN15832J TYPE OR EQUAL.
   IC7 IS A SN15830J TYPE OR EQUAL.
6. A. NO STRAP CONNECTIONS FOR 75 BAUD.
   B. STRAP N TO 11 FOR 37.5 BAUD
   C. STRAP N TO 11, 15 TO K, AND 6 TO 12 FOR 50 BAUD
7. N.C. DENOTES NO CONNECTION



OR EQUAL.
QUAL.
QUAL.
YPE OR EQUAL.
EQUAL.
OR 75 BAUD.
UD
6 TO 12

Fig. 31. A1 Assembly (Ref 1:6-2)



NOTES:

1. UNLESS OTHERWISE INDICATED, RESISTANCES ARE IN OHMS, 1/4 W, ±5%. CAPACITANCES ARE IN PF.
2. ALL DIODES ARE IN4148, UNLESS OTHERWISE INDICATED.
3. ALL CHIPS: PIN 7 IS GND, PIN 14 IS +5V.
4. FOR CMPNT BD ASSY SEE DWG NO. D80281540
5. INTEGRATED CIRCUITS:
   IC5 AND IC8 ARE SN158093J TYPE OR EQUAL.
   IC2 AND IC7 ARE SN15836J TYPE OR EQUAL.
   IC1, IC3, IC6 AND IC9 ARE SN15846J OR EQUAL.
5. CONT
   IC10 IS A SN15862J TYPE OR EQUAL.
   IC4 AND IC11 ARE SN15830J TYPE OR EQUAL.
6. * DENOTES STRAP OPTION.
7. A. STRAP 3 TO 4 FOR 75 BAUD.
   B. STRAP 3 TO 5 FOR 50 BAUD.
   C. STRAP 3 TO 6 FOR 325 BAUD
8. N.C DENOTES NO CONNECTION

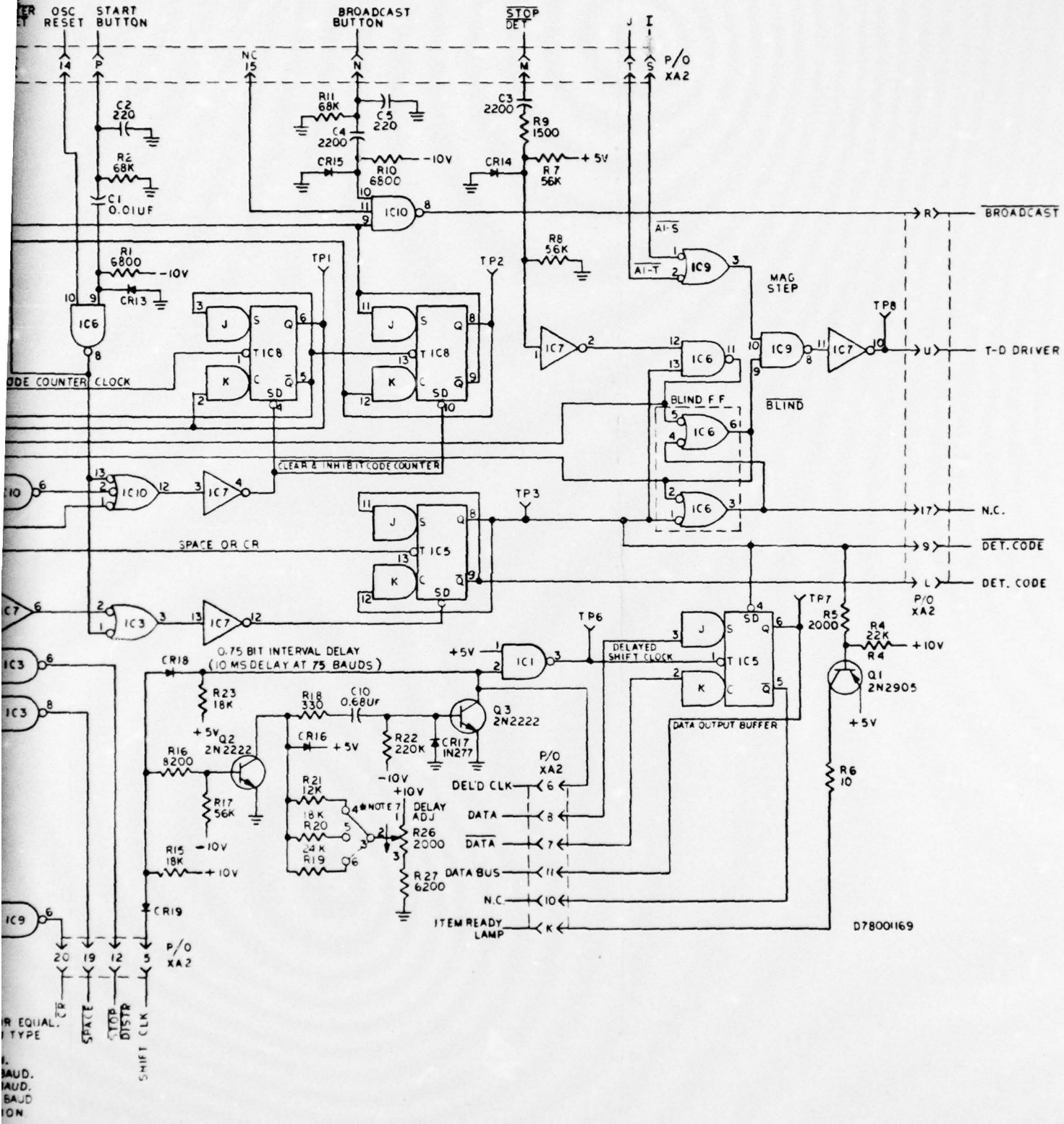
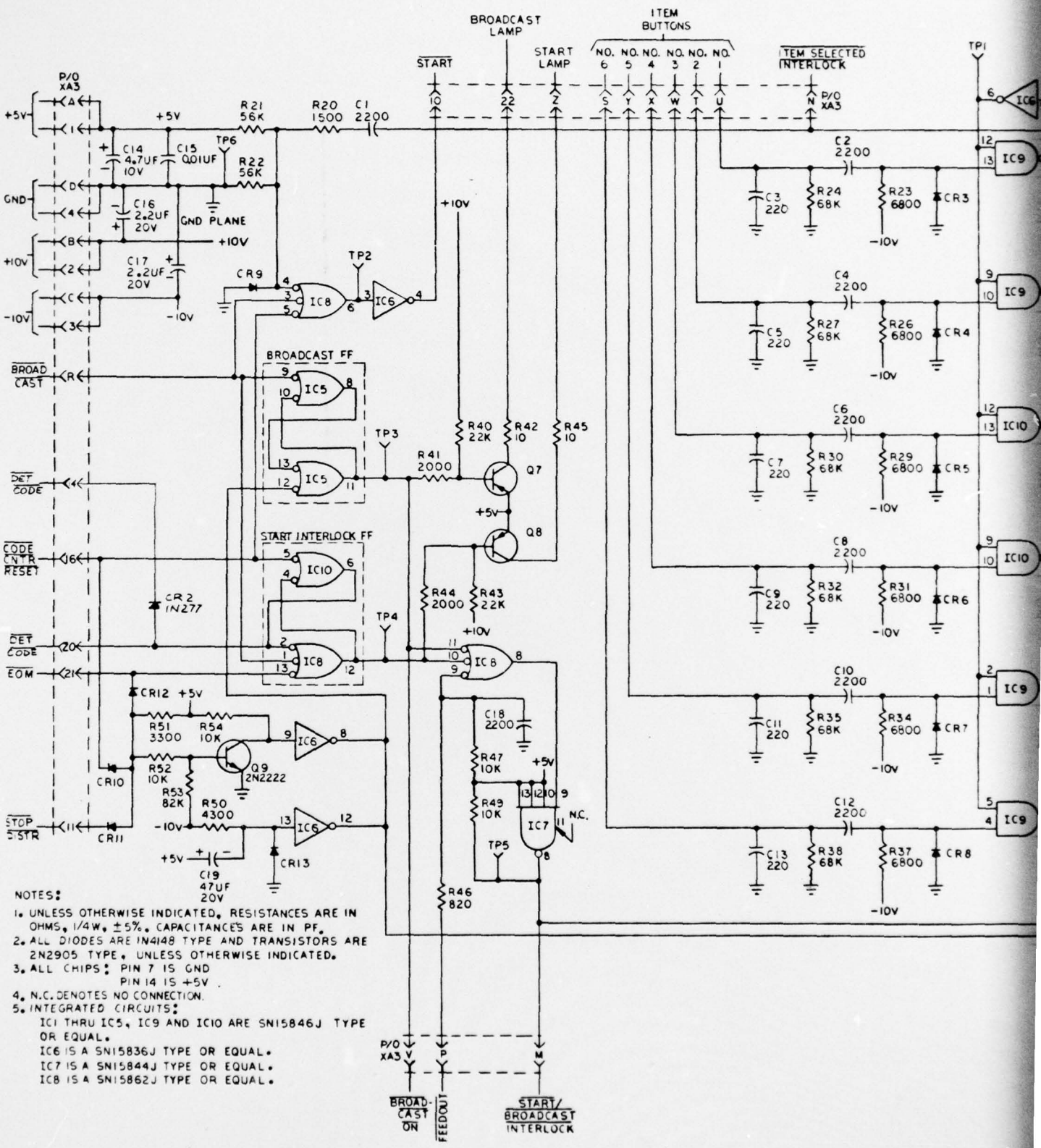


Fig. 32. A2 Assembly (Ref 1:6-3)



NOTES:

1. UNLESS OTHERWISE INDICATED, RESISTANCES ARE IN OHMS, 1/4W, ±5%. CAPACITANCES ARE IN PF.
2. ALL DIODES ARE IN4148 TYPE AND TRANSISTORS ARE 2N2905 TYPE, UNLESS OTHERWISE INDICATED.
3. ALL CHIPS: PIN 7 IS GND
   PIN 14 IS +5V.
4. N.C. DENOTES NO CONNECTION.
5. INTEGRATED CIRCUITS:
   IC1 THRU IC5, IC9 AND IC10 ARE SN15846J TYPE OR EQUAL.
   IC6 IS A SN15836J TYPE OR EQUAL.
   IC7 IS A SN15844J TYPE OR EQUAL.
   IC8 IS A SN15862J TYPE OR EQUAL.

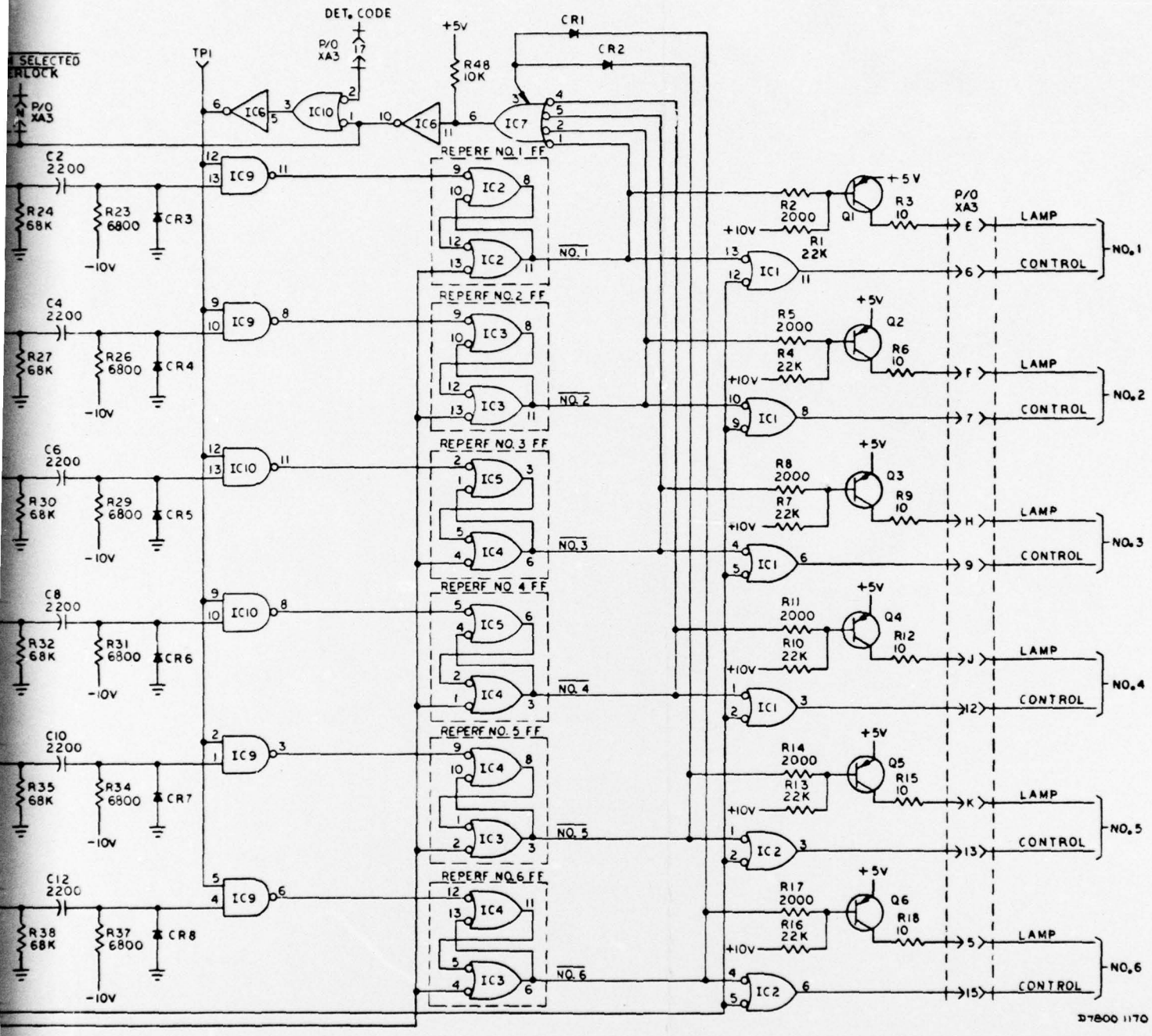
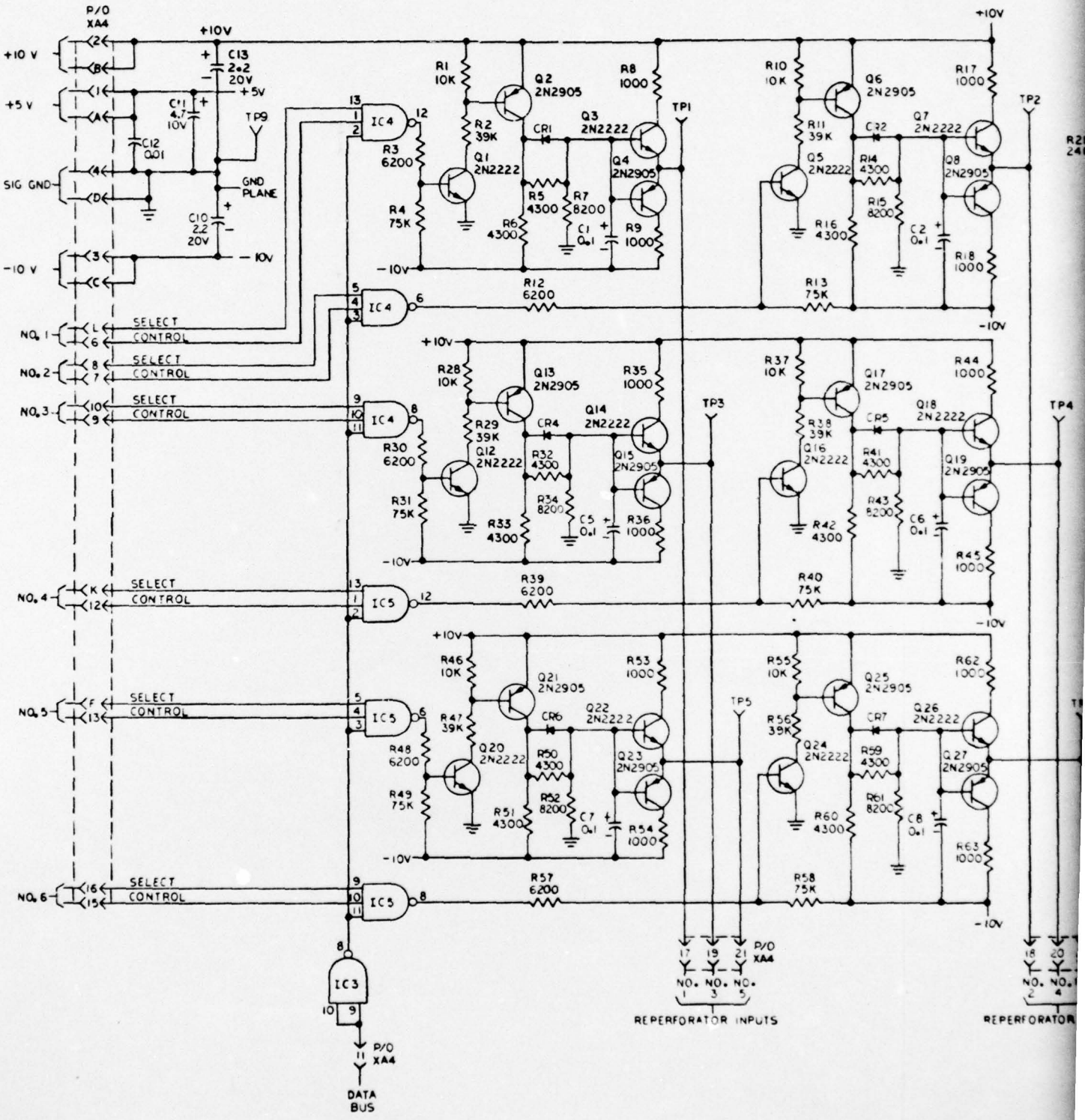
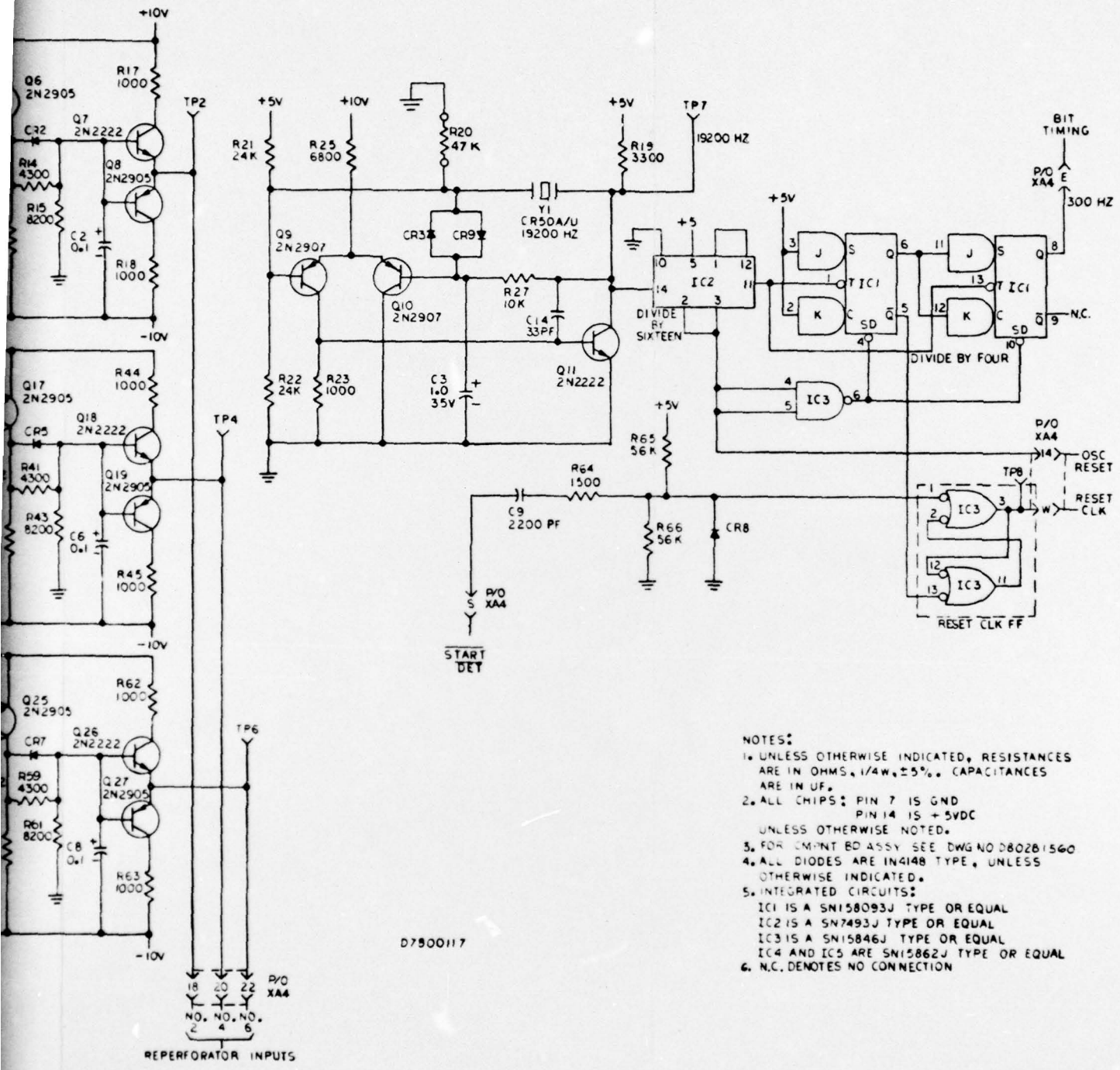


Fig. 33. A3 Assembly (Ref 1:6-4)


P/O XA4
+10 V
+5 V
SIG GND
-10 V
+10V
+5V
GND PLANE
-10V
C13 2.2 20V
C11 4.7 10V
C12 0.01
C10 2.2 20V
TP9
TP1
TP2
TP3
TP4
TP5
IC4
IC5
IC3
R1 10K
R2 39K
R3 6200
R4 75K
R5 4300
R6 4300
R7 8200
R8 1000
R9 1000
R10 10K
R11 39K
R12 6200
R13 75K
R14 4300
R15 8200
R16 4300
R17 1000
R18 1000
Q1 2N2222
Q2 2N2905
Q3 2N2222
Q4 2N2905
Q5 2N2222
Q6 2N2905
Q7 2N2222
Q8 2N2905
CR1
CR2
C1 0.1
C2 0.1
R28 10K
R29 39K
R30 6200
R31 75K
R32 4300
R33 4300
R34 8200
R35 1000
R36 1000
R37 10K
R38 39K
R39 6200
R40 75K
R41 4300
R42 4300
R43 8200
R44 1000
R45 1000
Q12 2N2222
Q13 2N2905
Q14 2N2222
Q15 2N2905
Q16 2N2222
Q17 2N2905
Q18 2N2222
Q19 2N2905
CR4
CR5
C5 0.1
C6 0.1
R46 10K
R47 39K
R48 6200
R49 75K
R50 4300
R51 4300
R52 8200
R53 1000
R54 1000
R55 10K
R56 39K
R57 6200
R58 75K
R59 4300
R60 4300
R61 8200
R62 1000
R63 1000
Q20 2N2222
Q21 2N2905
Q22 2N2222
Q23 2N2905
Q24 2N2222
Q25 2N2905
Q26 2N2222
Q27 2N2905
CR6
CR7
C7 0.1
C8 0.1
NO. 1
NO. 2
NO. 3
NO. 4
NO. 5
NO. 6
SELECT
CONTROL
P/O XA4
DATA BUS
REPERFORATOR INPUTS
REPERFORATOR



NOTES:

1. UNLESS OTHERWISE INDICATED, RESISTANCES ARE IN OHMS, 1/4W, ±5%. CAPACITANCES ARE IN UF.
2. ALL CHIPS: PIN 7 IS GND
   PIN 14 IS +5VDC
   UNLESS OTHERWISE NOTED.
3. FOR CMPNT BD ASSY SEE DWG NO D8028156O
4. ALL DIODES ARE IN4148 TYPE, UNLESS OTHERWISE INDICATED.
5. INTEGRATED CIRCUITS:
   IC1 IS A SN158093J TYPE OR EQUAL
   IC2 IS A SN7493J TYPE OR EQUAL
   IC3 IS A SN15846J TYPE OR EQUAL
   IC4 AND IC5 ARE SN15862J TYPE OR EQUAL
6. N.C. DENOTES NO CONNECTION

Fig. 34. A4 Assembly (Ref 1:6-5)

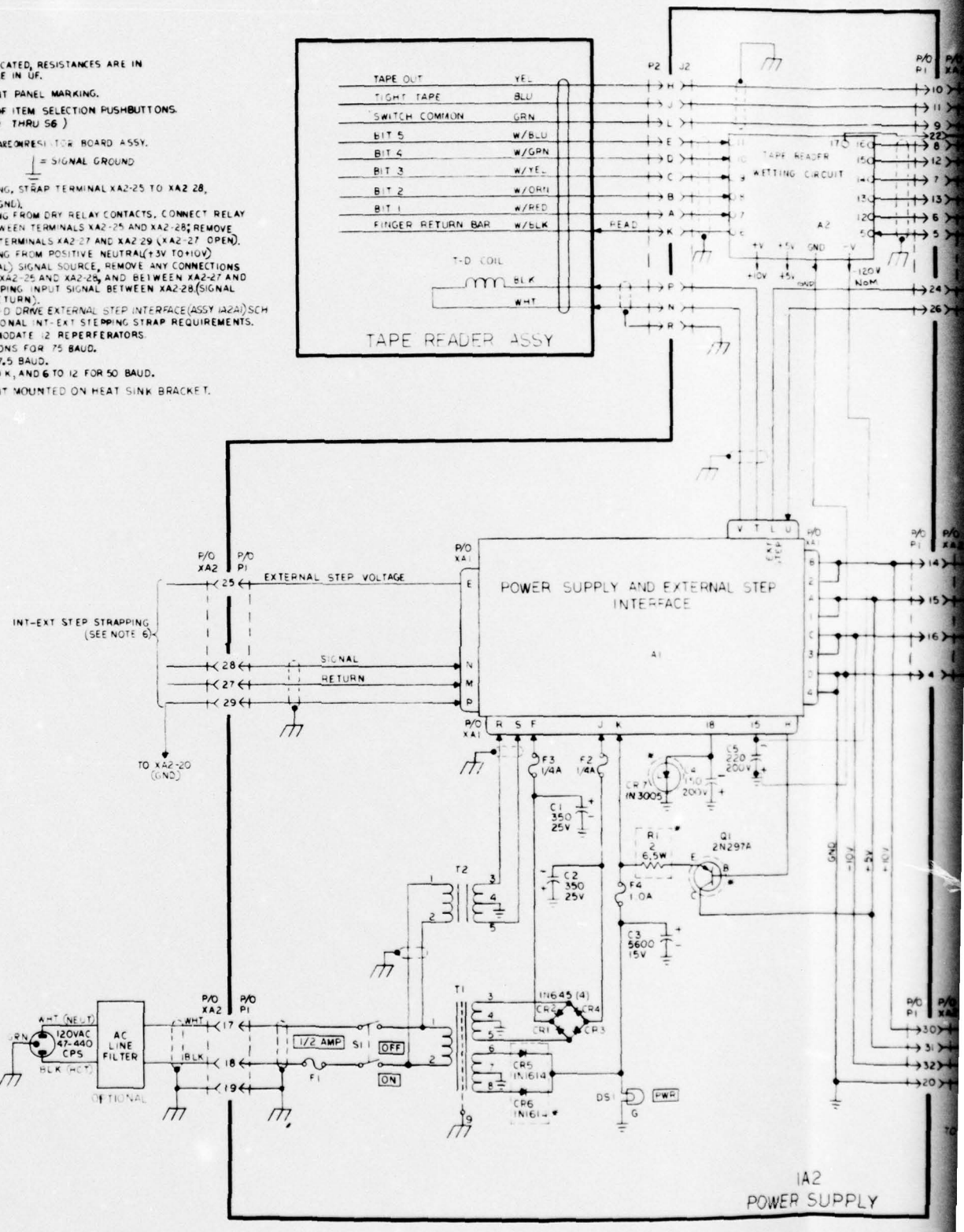
APPENDIX D

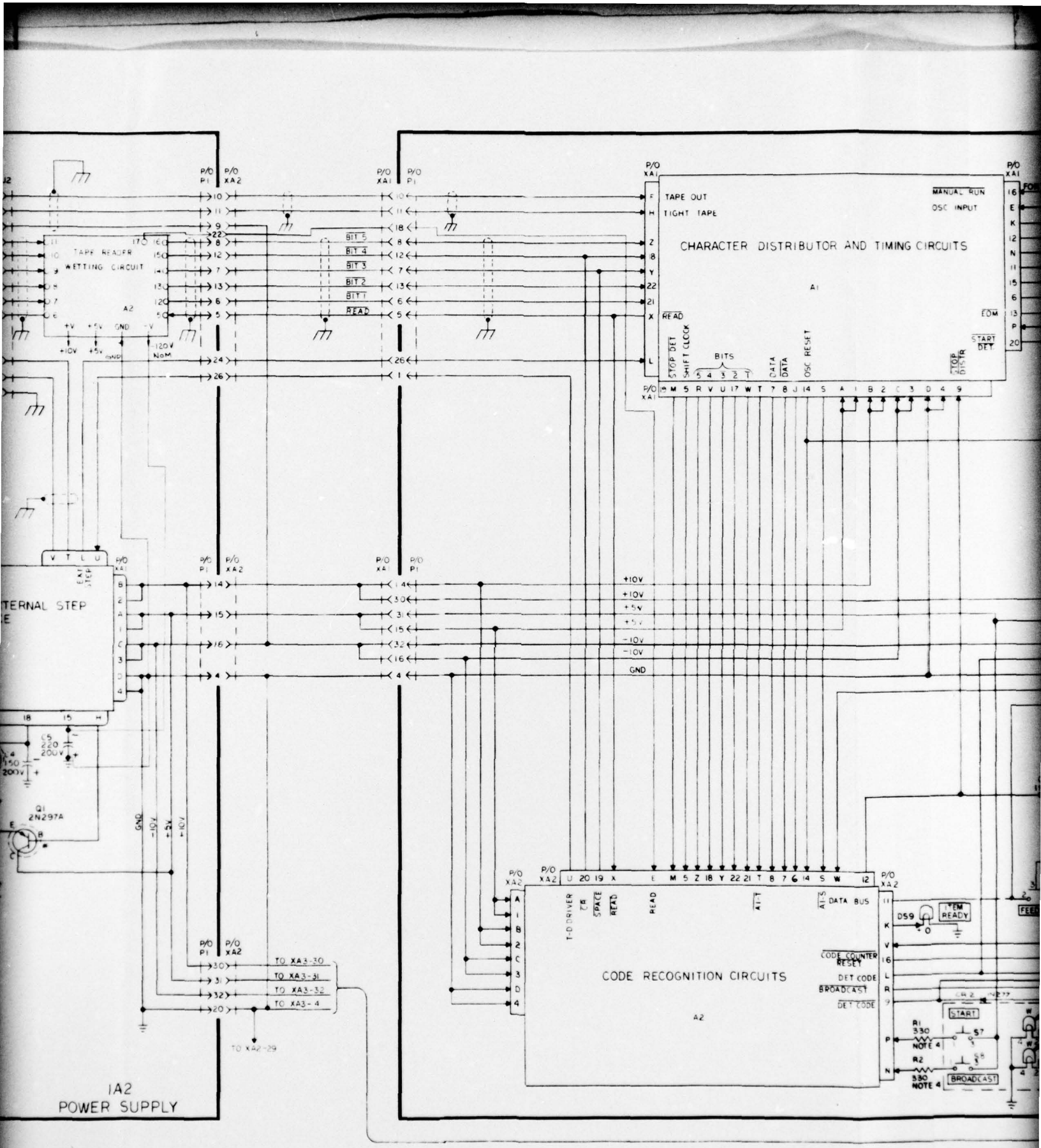
MAPU Schematic Interconnect Diagram

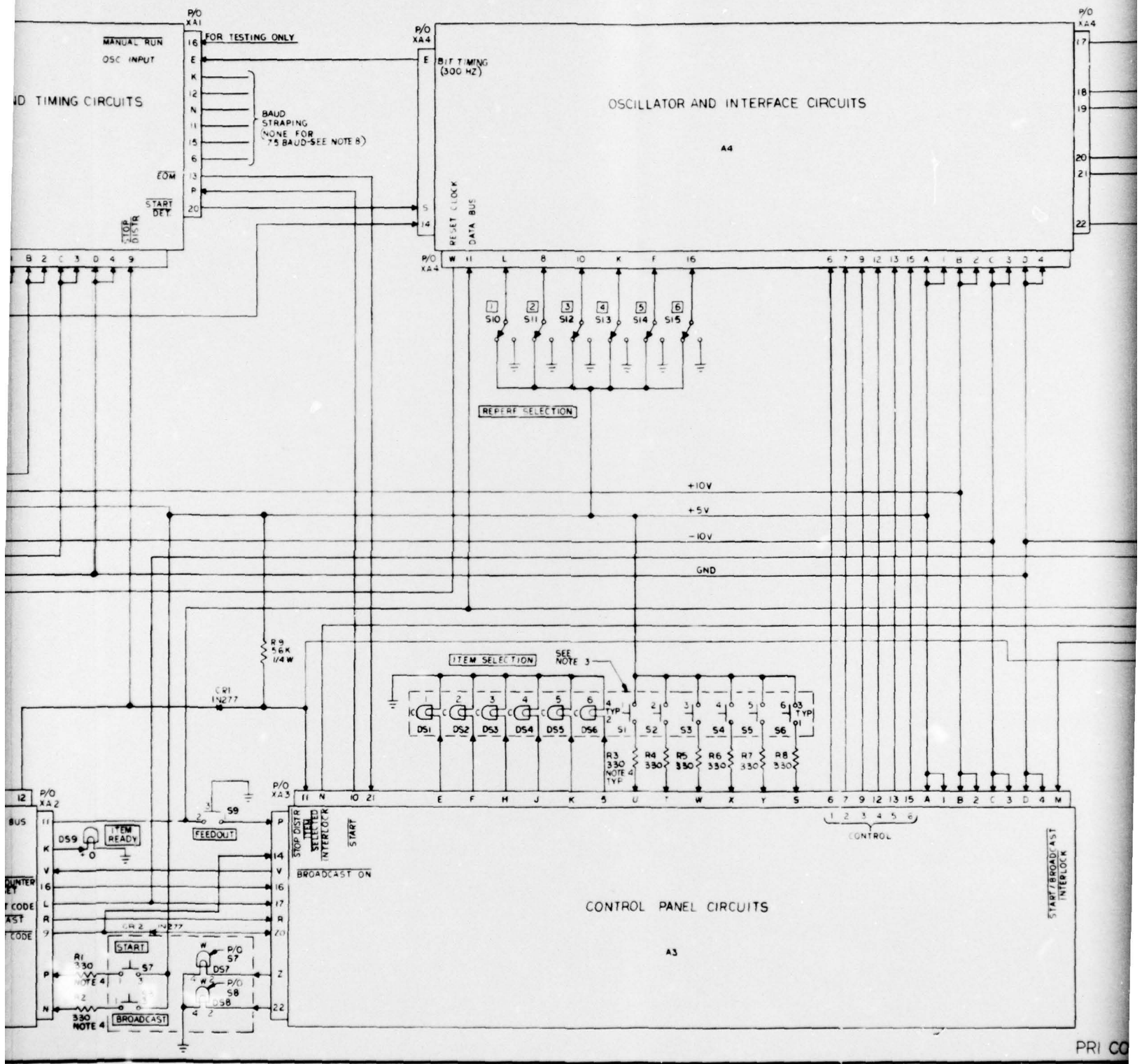
The diagram in Fig. 35 shows the schematic interconnection of the 1A1, 1A2, and optional 1A3 Assemblies.

NOTES:

1. UNLESS OTHERWISE INDICATED, RESISTANCES ARE IN OHMS. CAPACITANCES ARE IN UF.
2. [ ] INDICATES FRONT PANEL MARKING.
3. ITEM LAMPS ARE PART OF ITEM SELECTION PUSHBUTTONS. (DS1 THRU DS6 AND S1 THRU S6)
4. RESISTORS R1 THRU R8 ARE CONTAINED IN RESISTOR BOARD ASSY.
5. ⏚ = CHASSIS GROUND ⏚ = SIGNAL GROUND
6. A. FOR INTERNAL STEPPING, STRAP TERMINAL XA2-25 TO XA2-28, AND XA2-27 TO XA2-29 (GND).
   B. FOR EXTERNAL STEPPING FROM DRY RELAY CONTACTS, CONNECT RELAY CONTACTS (FORM A) BETWEEN TERMINALS XA2-25 AND XA2-28; REMOVE CONNECTION BETWEEN TERMINALS XA2-27 AND XA2-29 (XA2-27 OPEN).
   C. FOR EXTERNAL STEPPING FROM POSITIVE NEUTRAL (+3V TO +10V) OR POLAR (±6V NOMINAL) SIGNAL SOURCE, REMOVE ANY CONNECTIONS BETWEEN TERMINALS XA2-25 AND XA2-28, AND BETWEEN XA2-27 AND XA2-29. CONNECT STEPPING INPUT SIGNAL BETWEEN XA2-28 (SIGNAL INPUT) AND XA2-27 (RETURN).
   D. SEE POWER SUPPLY T-D DRIVE EXTERNAL STEP INTERFACE (ASSY 1A2A1) SCH DIAGRAM FOR ADDITIONAL INT-EXT STEPPING STRAP REQUIREMENTS.
7. USE ASSY 1A3 TO ACCOMODATE 12 REPERFERATORS.
8. A. NO STRAP CONNECTIONS FOR 75 BAUD.
   B. STRAP N TO 11 FOR 37.5 BAUD.
   C. STRAP N TO 11, 15 TO K, AND 6 TO 12 FOR 50 BAUD.
9. ★ DENOTES COMPONENT MOUNTED ON HEAT SINK BRACKET.




CHARACTER DISTRIBUTOR AND TIMING CIRCUITS
A1
TAPE OUT
TIGHT TAPE
MANUAL RUN
OSC INPUT
READ
STOP DET
SHIFT CLOCK
BITS
DATA
DATA
OSC RESET
STOP DISTR
START DET
EOM
TAPE READER
WETTING CIRCUIT
A2
BIT 5
BIT 4
BIT 3
BIT 2
BIT 1
READ
+10V
+5V
GND
-10V
-120V NOM
CODE RECOGNITION CIRCUITS
A2
T-D DRIVER
CR
SPACE
READ
READ
ATT
ATS
DATA BUS
CODE COUNTER RESET
DET CODE
BROADCAST
DET CODE
ITEM READY
DS9
START
BROADCAST
R1 330 NOTE 4
R2 330 NOTE 4
S7
S8
TO XA3-30
TO XA3-31
TO XA3-32
TO XA3-4
TO XA2-29
Q1 2N297A
C5 220 200V
1A2
POWER SUPPLY


MANUAL RUN
OSC INPUT
FOR TESTING ONLY
BAUD STRAPING (NONE FOR 75 BAUD-SEE NOTE 8)
EOM
START DET
STOP DISTR
BIT TIMING (300 HZ)
RESET CLOCK
DATA BUS
OSCILLATOR AND INTERFACE CIRCUITS
A4
REPERF SELECTION
S10
S11
S12
S13
S14
S15
+10V
+5V
-10V
GND
R9 5.6K 1/4 W
CR1 IN277
ITEM SELECTION
SEE NOTE 3
DS1
DS2
DS3
DS4
DS5
DS6
S1
S2
S3
S4
S5
S6
R3 330 NOTE 4 TYP
R4 330
R5 330
R6 330
R7 330
R8 330
CONTROL
ITEM READY
DS9
S9
FEEDOUT
STOP DISTR
ITEM SELECTED INTERLOCK
START
BROADCAST ON
START/BROADCAST INTERLOCK
CR2 IN277
START
S7
DS7
DS8
R1 330 NOTE 4
330 NOTE 4
BROADCAST
CONTROL PANEL CIRCUITS
A3

3

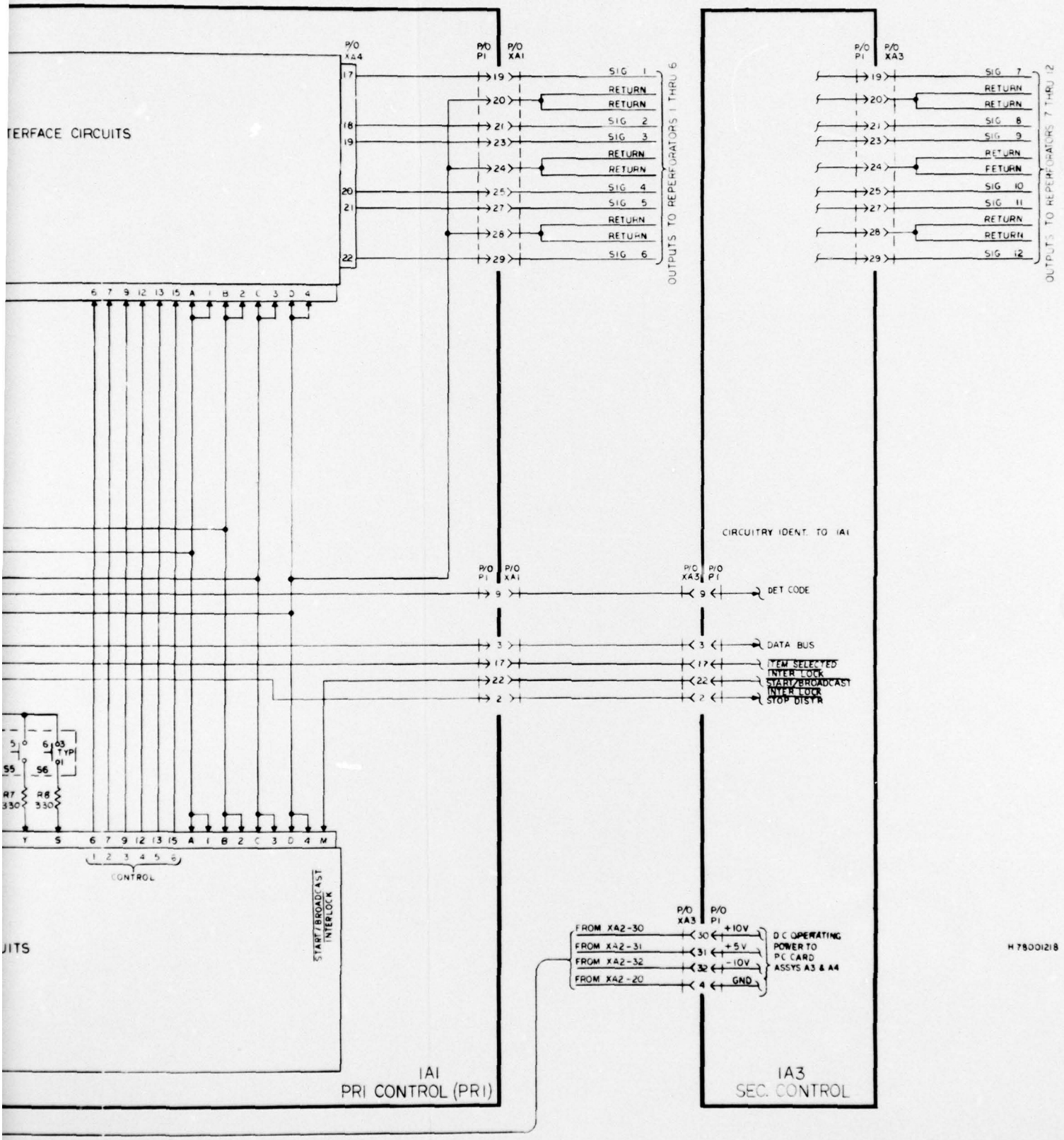


Fig. 35. MAPU Schematic Diagram (Ref 1:6-8)

VITA

Robert G. Shively was born on 31 May 1949, in Los Angeles, California. He graduated from high school in Encino, California in 1967 and received an appointment to the United States Military Academy where he earned the Bachelor of Science degree in June of 1971. Upon graduation he was commissioned a 2LT in the Regular Army. He attended both the Basic and Advanced Signal Officers Course at Fort Gordon, Georgia, and the Communications-Electronics Staff Officers Course at Fort Sill, Oklahoma. He served three years in Europe as a member of the 8th Signal Battalion, 8th Infantry Division. In September of 1975 he entered the School of Engineering, Air Force Institute of Technology.

Permanent Address: 5952 Jellico Avenue
Encino, California 91316

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: GE/EE/77-3	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)*: MODIFICATION OF THE AN/TGC-26 MULTIPLE ADDRESS PROCESSING UNIT TO ACCEPT JANAP 128 FORMAT	5. TYPE OF REPORT & PERIOD COVERED: MS Thesis 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Robert G. Shively, Capt. SigC	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Air Force Institute of Technology (AFIT/EN), Wright-Patterson AFB, Ohio 45433	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: AFCS/DOOT, Richards Gebaur AFB, MO 64030	12. REPORT DATE: March 1977 13. NUMBER OF PAGES: 118
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*	15. SECURITY CLASS. *(of this report)*: UNCLAS 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

Approved for public release;
IAW AFR 190-17
JERRAL F. GUESS, Captain USAF
Director of Information

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

407L
JANAP 128
AN/TGC-26
Torn tape relay
Multiple Address Processing Unit (MAPU)

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

The Multiple Address Processing Unit (MAPU) of the AN/TGC-26 tape relay center was originally designed to process ACP 127 message format. With the advent of JANAP 128 as a standardized message format a pressing need was felt for a MAPU modification to allow processing of this format. Under the sponsorship of AFCS/DOOT, a partial MAPU was assembled at the Air Force Institute

over

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

of Technology, and the code recognition logic circuitry was redesigned. Software simulations of the proposed redesign were tested using the Digital Logic Simulator program on the CDC CYBER computer system. A final prototype version of the modified MAPU was successfully tested at Wellesley Air National Guard Station, Massachusetts, using the personnel and equipment of the 253rd CMBTCG. A final schematic diagram and production recommendations are presented herein.